Event-based Neuromorphic Stereo Vision

Dissertation zur Erlangung der naturwissenschaftlichen Doktorwürde
(Dr. sc. nat.)

vorgelegt der

Mathematisch-naturwissenschaftlichen Fakultät

der

Universität Zürich

von

Marc Osswald

von

Basel-Stadt, BS

Promotionskomittee

Prof. Dr. Giacomo Indivieri (Vorsitz)
Prof. Dr. Tobias Delbruck
Prof. Dr. Ryad Benosman

Zürich 2016

We must use time as a tool
not as a couch.
— John F. Kennedy

To my parents...

Disclaimer

I hereby declare that the work in this thesis is that of the candidate alone, except where indicated in the text, and as described below.

Chapters 2 and 3 contain reviews of relevant work from different research disciplines. The work described in these chapters was conducted by others, as indicated by the corresponding references.

Chapter 4 is partially based on ideas and work that was conducted in collaboration with Ryad Benosman and Sio-Hoi Ieng in the course of a three-month visit at the Institut de la Vision in Paris, France.

The stereo network described in Chapter 5 emerged from the idea of applying a cooperative process to the data provided by event-based temporal contrast sensors. Recent work describing similar approaches has revealed that there is some doubt about who the original idea should be credited to. Mahowald (1994a) was the first to use a cooperative process for stereo vision together with silicon retinas. That work served as the main inspiration for the present thesis. Here, the idea was reformulated in the context of event-based temporal contrast sensors, as first proposed in Osswald (2011). This proposal was made without prior knowledge of a similar idea that had already been published in a Semester Project (Hess, 2006). Independently from the work described here, others have proposed similar ideas while this project was being carried out (Piatkowska et al., 2013, 2014; Firouzi and Conradt, 2015). Nevertheless, despite the common origin, the work described here deviates substantially from others, as discussed in the relevant sections.

Some of the experiments in Chapter 6 were implemented on a neuromorphic processor that was primarily developed for purposes other than stereo vision. The sections 6.2.1 and 6.2.2 explaining the architecture and relevant building blocks of the chip were taken from the original publication (Qiao et al., 2015). The use of “we” in these sections refers to the authors of that publication.

Acknowledgements

I would like to express my gratitude to several people who have supported me during this project and without whom this thesis would not have existed. I would like to express particular gratitude to my advisor, Prof. Giacomo Indiveri, for his patience, motivation and immense knowledge and for giving me the opportunity to carry out this project. I would also like to thank the rest of my thesis committee: Prof. Tobi Delbrück for his insightful comments and encouragement and Prof. Ryad Benosman for the wonderful time I spent in Paris, where some of the main concepts of this research were formed. Besides my committee, I am also grateful to Dr. Daniel Kiper for his valuable feedback.

My sincere thanks goes to my colleagues, Federico Corradi, Hesham Mostafa and Qiao Ning, with whom I have developed and tested two generations of neuromorphic processors, Jonathan Binas, Daniel Fasnacht and Richard George, with whom I have designed PCBs and programmed software, Lorenz Müller and Julien Martel, for enlightening me in theories and algorithms, and the entire NCS group, for fruitful discussions and feedback. I also wish to thank all of my friends and collaborators at the *Institut de la Vision* in Paris, in particular Sio-Hoi Ieng, Joao Carneiro and Xavier Lagorce.

I would like to thank the people from *iniLabs*, Sim Bamford, Luca Longinotti and Vicente Villanueva for their great support with the sensor hardware.

I am very grateful to the *Capo Caccia Neuromorphic Engineering Workshop*, where I participated annually from 2012 to 2015. This unconventional and exciting workshop, where all of the world's experts from the community are brought together, was definitely one of the major highlights of my project. I want to thank all the people with whom I have collaborated, for the sleepless nights through which we worked together to create a functioning demonstration, for the peaceful afternoons we spent at the beach to recover from the hacking marathons, and for the glamorous evenings where we hatched earth-shattering ideas at the bar. These unique experiences made it possible to conduct this research and ultimately led to the inception of our start-up company *Insightness*, which I co-founded together with Christian Brändli and Tobi Delbrück.

I would also like to place on record my sense of gratitude to Eoin Jones, for proofreading this thesis.

Last but not least, I would like to thank my parents, brothers and sister for their moral support throughout the entire duration of this project and in my life in general.

Zurich, 2016 M. O.

Abstract

Depth perception from stereo vision is a challenging problem, affecting both biological and artificial vision systems. It involves finding correspondences in the different views captured by the visual sensors. While animals solve this problem effortlessly, the field of machine vision has struggled for many years to find algorithms and strategies that can be as efficient and robust as those used in biological nervous systems. One major difference between biological visual systems and conventional machine vision systems, which is partly responsible for this large gap in performance, lies in the vision sensors themselves. Biological vision systems carry out self-timed and continuous sensing of the visual scene. Conversely, the use of frame-based cameras has been prevalent in the field of machine vision, which capture static images at regular time intervals. Machine stereo vision algorithms have been optimized to extract depth information from pairs of static images, typically acquired at frequencies of a few tens of Hertz. As a result, the most common approaches suffer from a trade-off between latency and computational cost, which is caused by the processing of redundant data from subsequent images. In recent times, however, a new class of event-based vision sensor has become available. These sensors are called silicon retinas because they are based on models of the mammalian visual retina. The silicon retina used in this study produces continuous streams of spikes (or events), that only encode the changes caused by movement in a scene, thus providing a form of visual output that is sparse and excludes redundant information. These new sensors now offer the possibility to implement efficient frame-less machine vision algorithms that are much more closely related to those of real biological systems.

This thesis proposes an algorithm for extracting depth information from silicon retinas arranged in a stereo setup. The algorithm exploits the fine structure of temporal information extracted by the silicon retina and processes it in a self-timed and data-driven manner inspired by one of the main computational principles of the brain. An analysis of the central operational concept of the algorithm leads to the proposal of a spiking neural network model to extract depth information from silicon retinas. The neural network is shown to robustly solve the stereo correspondence problem. The proposed model builds upon well-grounded theories of disparity processing in the mammalian brain, and it confirms neurophysiological findings related to stereo vision. In particular, the model proposes an explanation for how visual motion promotes depth perception. This casts new light on the unresolved question of how the correspondence problem is solved in the brain. At the same time, the model can be applied to very concrete and practical machine vision problems. The stereo model is implemented on custom-made, mixed analog and digital hardware, with the ultimate aim of

developing an artificial stereo vision system that outperforms current machine stereo vision systems and paves the way for the smart devices of the future.

Zusammenfassung

Tiefenwahrnehmung durch Stereosehen ist ein anspruchsvolles Problem, sowohl für Menschen als auch für Maschinen. Die Hauptschwierigkeit liegt darin, Übereinstimmungen zwischen den verschiedenen Blickwinkeln verschiedener Sensoren zu finden. Während Menschen und Tiere diese Aufgabe scheinbar mühelos lösen können, sucht die Forschung im Bereich des maschinellen Sehens seit Jahren nach entsprechenden Algorithmen und Strategien, die so effizient und robust funktionieren, wie jene biologischer Systeme. Ein Hauptunterschied zwischen biologischen und herkömmlichen Bildverarbeitungssystemen, der teilweise für diese Diskrepanz verantwortlich ist, hat mit der funktionsweise der Sensoren zu tun. Biologische Sensoren (z.B. das menschliche Auge) erfassen visuelle Szenen kontinuierlich und kommunizieren Informationen selbständig. Dadurch generieren die in Maschinen eingesetzten Kameras statische Bilder durch Abfragen in regelmässigen Zeitabständen. Computer-Algorithmen wurden hauptsächlich für das Bearbeiten von solchen statischen Bildern, welche typischerweise mit Frequenzen von ein paar Dutzend Hertz erfasst werden, entwickelt. Dieser Ansatz hat den Nachteil, dass jeweils die ganzen Bilder, welche oft einen grossen Anteil an redundanten Information enthalten, verarbeitet werden müssen. Das führt zu einem erhöhten Rechenaufwand und kostet Zeit und Enerige. In jüngster Zeit ist jedoch eine neue Art von “ereignisgesteuerten” Kameras verfügbar geworden. Diese Sensoren werden auch *Silicon Retina* genannt, weil sie ähnlich funktionieren wie das menschliche Auge. Die in dieser Dissertation verwendete *Silicon Retina*, erfasst nur die visuellen Änderungen im Blickfeld. Also hauptsächlich bewegte Objekte aber auch jene Unterschiede, die sich etwa durch einen Perspektivenwechsel ergeben. So wird eine kontinuierliche Ausgabe von Ereignissen erzeugt, basierend auf jeder einzelnen Veränderung im Erfassungsbereich. Dadurch wird redundante Information reduziert und die zeitliche Auflösung ist maximal. Diese Sensoren ermöglichen es, neue und effizientere Computer-Algorithmen zu implementieren, welche keine statischen Bilder mehr benötigen, sondern direkt visuelle Ereignisse verarbeiten, und somit dem Verarbeitungsprozess im menschlichen Gehirn besser entsprechen.

Diese Arbeit stellt einen Algorithmus zur Extraktion von Tiefeninformation aus ereignisgesteuerten, stereoskopischen Kameras vor. Der Algorithmus nutzt die zeitlich hochaufgelöste Struktur der visuellen Ereignisse und verarbeitet sie nach dem effizienten, ereignisgesteuertem Prinzip der Informationsverarbeitung im menschlichen Gehirn. Eine tiefgründige Analyse des zugrunde liegenden Konzepts des Algorithmus führt zu einem Modell eines *neuronalen Netzwerks*, welches die robuste und exakte Tiefenwahrnehmung mittels ereignisgesteuerten Kameras implementiert. Das vorgeschlagene Modell basiert auf fundierten Theorien über den

Prozess der Tiefenwahrnehmung im Gehirn, und bestätigt neurophysiologische Forschungsergebnisse zum Stereosehen. Insbesondere, liefert das Modell eine Erklärung, wie die Erfassung von Bewegungen die Tiefenwahrnehmung im Gerhin begünstigen kann. Dies wirft ein neues Licht auf die ungelöste Frage, wie die Aufgabe des Stereosehens im Gehirn gelöst wird. Gleichzeitig kann das Modell auf konkrete und praktische Anwendungen des maschinellen Stereosehens angewandt werden. Dafür werden analoge und digitale, elektrische Schaltungen untersucht, welche der Funktionsweise des Gehirns nachempfunden wurden. Das hat zum Ziel, ein künstliches System zu entwickeln, das herkömmliche Ansätze des maschinellen Stereosehens übertrifft und damit den Weg für zukünftige, intelligente Sensoren bahnt.

Contents

List of Figures

List of Tables

1 Introduction

For many living beings, visual perception is by far the most important way of acquiring and processing information from their environment. In performing visual tasks, such as navigation, even the visual systems of many insects outperform state of the art computers in terms of speed and precision, requiring far less computational might and consuming only a fraction of the power. The main reason for this performance difference lies within the structure of the hardware itself. Traditional computers are digital, centralized and process information sequentially at relativistic speed. Neural circuitry, on the other hand, consists of networks of massively parallel connected units, propagating signals at very low speed.

An essential function of the biological vision systems of higher organisms is depth perception. Visual depth perception can be derived from motion (motion parallax) cues, perspective cues, oculomotor cues (e.g. accommodation), lighting and shading cues, interpositional (occlusion) cues or binocular cues (Howard, 2002). The latter is exclusively used by organisms which require fast and precise depth perception, such as predators. The key principle involved in depth perception derived from binocular cues is *stereopsis*. In humans and many mammalians the eyes are horizontally spaced, which results in two slightly different projections of a scene in the retinas. The brain is able to register and process these differences in order to perceive depth. This process is called stereopsis. Stereopsis is subject to a difficult problem, commonly known as the *stereo correspondence problem* (Hartley and Zisserman, 2004), which refers to the complexity of finding visual correspondences among different views. Over the course of evolution, a wide variety of biological stereoscopic vision systems have emerged, which are specific to an animal's anatomy, environment or behavior. Despite their differences, they are all subject to the stereo correspondence problem and they all found a way to solve it. Assuming there is a common neural architecture which underlies the foundation for solving the correspondence problem, the aim of this thesis is to understand the organizing principles of this architecture and make use of them in an artificial system. It is well known that the processing of visual information in the brain already starts at the level of the retina. Thus, the whole vision system, ranging from the retinas to the neural circuitry which solves the correspondence problem, is considered to be the architecture under investigation.

Stereopsis has also been studied in the context of artificial systems, particularly in the field of machine vision where it is referred to as *machine stereo vision*. However, the approach taken in such studies is very different to that of the brain. In the field of machine vision, visual information is generally represented in the form of static frames (or images) captured at regular intervals. This information is processed using a *frame-based* computing paradigm which is inefficient, as subsequent frames carry a lot of redundant information. This paradigm also lacks *precise temporal dynamics* because frames and pixels are sequentially processed in a manner which is governed by an external clock rate. These factors are all constraints that are not found in biological systems. It stands to reason that aspects of biological vision systems such as *temporal* dynamics, *self-timed* computation and *parallel* architectures are key features of stereopsis. This thesis will investigate how those key principles can be adopted in order to solve the stereo correspondence problem in a more biologically-inspired manner using dedicated, so-called *neuromorphic* hardware.

In the 1980s, the idea of embodying neural computation in silicon *analog very large-scale integrated* (aVLSI) circuits emerged. Collectively, such circuits were referred to as neuromorphic hardware (Mead, 1989). Early successes in emulating the sensory periphery of neural systems led to an initial euphoria and the field expanded rapidly. In the meantime, many similar systems have been developed, particularly for perceptual tasks. One of the most popular contributions of the field has been the development of the *silicon retina*, an analog integrated circuit that processes visual stimuli similarly to the human retina (Mahowald and Douglas, 1991). A whole series of vision chips followed, involving spatial and temporal contrast sensors, motion detectors, optical flow and even stereo correspondence chips (Moini, 2000). While these achievements were striking, the majority served mainly as proof of a deeper scientific understanding of retinal circuitry, but only few had any practical significance.

However, new approaches are now emerging which are beginning to have practical significance (Lichtsteiner et al., 2008; Lenero-Bardallo et al., 2010; Posch et al., 2010a; Farabet et al., 2011; Khan et al., 2008; Merolla et al., 2014). Building on these significant developments, this thesis aims to work towards a neuromorphic architecture by implementing a biologically-inspired stereo vision system. This system will primarily serve to gain insight into the feasibility of applying the key principles of stereopsis to artificial systems, with the aim of improving their efficiency.

1.1 Stereo vision at a glance

This thesis is a highly interdisciplinary form of research which combines knowledge and practice from the fields of computer science and neuroscience. One of the goals of the thesis is to unify these fields at the interface of stereo vision. Thus, a comprehensive and distinct terminology is required.

The field of vision involves everything related to the sensing and processing of visible light. A general distinction between **biological vision** and **machine vision** can be made, in that

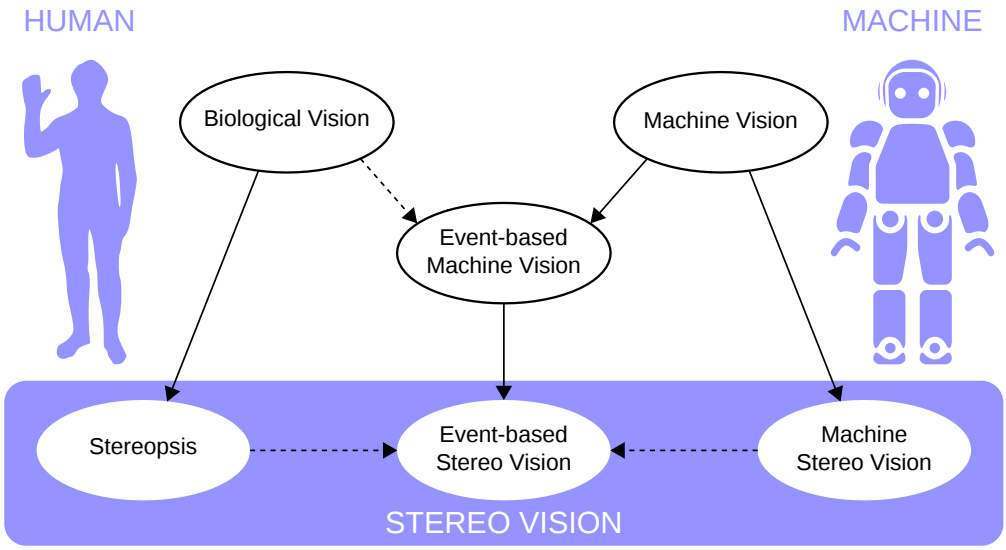


Figure 1.1: Etymology and classification of stereo vision.

biological vision refers to vision by living beings whereas machine vision collectively refers to vision by any artificial system (such as computers, machines or robots for example). At the interface of biological and machine vision is the field of **event-based machine vision**. Although strictly speaking, as implied by the name, event-based machine vision falls under the rubric of machine vision, it is inspired by concepts from biological vision. The term "event-based" describes a computing paradigm, which concerns the way information is processed. As opposed to a *frame-based system*, an *event-based system* processes information at the time it occurs rather than at fixed intervals governed by an external clock rate. More generally, the terms **data-driven**, **self-timed** or **asynchronous** describe the same concept, whereby the latter is often used in reference to digital logic circuits. Conversely, the terms **clocked** and **synchronous** are also used to describe frame-based systems. The subfield of event-based machine vision that deals with stereo vision is called **event-based stereo vision**. Similarly, while the field that deals with biological stereo vision is called **stereopsis,** the related subfield in machine vision is termed **machine stereo vision**. Collectively, these terms are unified in the field of **stereo vision**. A diagram illustrating the derivative roots of the different fields of stereo vision is shown in Fig. 1.1.

Many terms that are typically used in either computer science or neuroscience have a more general meaning in this thesis. A **processor** can be either a biological device, such as a neuron, or an artificial device like the CPU of a computer. Similarly, a **sensor** can be either biological or artificial. In this regard, the eye is referred to as a biological visual sensor while a video camera is an artificial or machine vision sensor. The terms *visual sensor* and *vision sensor* are used interchangeably. Thus, an **event-based camera** is a machine vision sensor that provides data in the form of a dynamic visual stream of events rather than static images. Event-based cameras, also called *silicon retinas*, are covered in Chapter 3.3. Unfortunately, it is not always possible to use unambiguous terminology. For instance, a neuron, which primarily denotes a

cell in the nervous system, is also used to describe a unit from an artificial neural network or a digital circuit that imitates the behavior of a real neuron. In these cases, however, the intended meaning should be clear from the context.

1.2 Bridging the gap from stereopsis to machine stereo vision

The neurons that code relative depth in the mammalian visual system were discovered about 50 years ago (Barlow et al., 1967). At that time, the study of depth perception from binocular vision (i.e., stereopsis) was a research area which mainly involved the fields of psychology, physiology, and neuroscience. Machine vision was an emerging area in the computer science domain, chiefly concerned with defining computer algorithms to analyze and extract information from static images. As the scope of machine vision expanded to address multiple images — obtained from stereo setups, for example — computer scientists developed algorithms inspired by stereopsis studies carried out in the fields of psychology and neuroscience. This was a highly interdisciplinary effort that led, for example, to one of the most popular algorithms proposed in the late 1970s to compute stereo correspondence: the *cooperative-competitive network* developed by Marr and Poggio (1976). Due to rapid progress in computing and vision sensor technologies, however, machine vision research departed from this interdisciplinary approach. As a result, the field of machine stereo vision developed independently, focusing mainly on solving the stereo correspondence problem and creating precise and detailed 3D reconstructions from pairs of static images. Meanwhile, key discoveries and progress were separately made in the domain of neuroscience. To list just a few examples of this progress, the cortical pathways involved in depth perception were discovered, disparity detectors were classified and modeled (Ohzawa et al., 1990), and the functional links between disparity-tuned cells and motion-in-depth were outlined (see Chapter 2.2 for a detailed review).

With the emergence of the field of "neuromorphic engineering" in the late 1980s, researchers started to emulate neural processing systems using electronic circuits, and to implement the principles of neural computation using artificial electronic systems based on very large scale integration (VLSI) technologies. Following this highly interdisciplinary approach, Mahowald (1994a) built an analog VLSI chip that successfully solved the stereo correspondence problem for one dimensional artificial retinal images using an architecture that was compatible with both machine vision algorithms and stereopsis models based on the anatomy of the primate visual system. This pioneering work yielded insights for both computer science and neuroscience, and is a perfect example of the potential of such multi-disciplinary research.

Advances in both VLSI technology and computational neuroscience have enabled neuromorphic engineers to develop powerful retina-like, two-dimensional vision sensors. These sensors adopt the key principles of neural processing systems, by transmitting continuous streams of spikes, rather than discrete sequences of image frames (Lichtsteiner et al., 2008; Posch et al., 2011). In these silicon retinas, each pixel produces an address-event (i.e., a spike labeled with the pixel's address) when the contrast sensed by that pixel changes by an amount

greater than a set threshold. All of the pixels in the sensor are independent and send events asynchronously. While the retinas produce no output in the case of static scenes, in the case of dynamic scenes, nearby retinas produce correlated outputs. This technology has led to an increased interest in studying stereo vision using event-based representation. Recently, several methods and algorithms have been proposed for such sensors, but most of them are biased, in one way or another, by the frame-based approach typically adopted in classical machine vision research. Furthermore, event-based algorithms are typically executed on a Von Neumann architecture, which is characterized by distinct and separate memory and processor units, and by computing styles that are very different from the way information is processed in biological neural systems.

This project aims to move away from the classical computer science mindset and create an event-based computing paradigm inspired by some of the key computational principles found in neuroscience research, such as spiking neurons. In doing so, the aim is to develop a simple algorithm that solves the correspondence problem while remaining well connected to established models of stereopsis. In addition, this algorithm is designed so that it can be directly mapped onto massively parallel neuromorphic computing platforms. Thus, this thesis aims to provide a unifying framework for both stereopsis and machine stereo vision research.

1.3 Organization of the thesis

The thesis contains three parts. The first part, comprising this chapter and Chapters 2 and 3, formulates the scientific problem under investigation, reviews the related fields, and introduces relevant concepts, materials and methods. The second part, comprising Chapters 4, 5 and 6, describes the main work carried out during this PhD project, which aimed to solve the scientific problem stated in the first part. Each chapter ends with a discussion of its content. Finally, Chapter 7 summarizes the project, draws some conclusions and makes suggestions about potentially fruitful areas of future research. The content of the chapters which follow is summarized below.

Chapter 2 introduces the fundamental problem of stereo vision, the *stereo correspondence problem*. The physiology of stereopsis is reviewed with a focus on the mechanisms that are relevant for this thesis. Subsequently, the focus shifts to the field of machine stereo vision, whereby existing cooperative algorithms for stereo correspondence are reviewed, followed by an outline of various approaches to machine stereo vision. Finally, the more recent trend towards real-time implementation on embedded systems is discussed in the context of the aims of this thesis.

Chapter 3 introduces the idea of brain-inspired computation and the field of neuromorphic engineering. The concept of event-based cameras is explained with an emphasis on *event-based temporal contrast sensors*, which were used in this project. Furthermore, neuromorphic processors, which form a substrate for event-based computation, are discussed. Within this context, the neuromorphic circuits that were used and partially designed as part of this thesis

are explained. Finally, the processing of event-based visual information is covered by reviewing a novel field of study called *event-based machine vision*. A categorization of event-based stereo vision algorithms and systems is presented. This categorization provides an overview of the existing research related to this thesis and illustrates why the research presented here could make a significant contribution to the field.

In Chapter 4, existing concepts of space-time representation of visual information and sampling strategies are investigated. This results in an ideal vision sensor being proposed. This ideal sensor is brought into context with the event-based temporal contrast sensors introduced previously. A useful conceptualization of temporal contrast events is introduced, on the basis of which an innovative event-based stereo vision algorithm is developed. The event-based representation of visual information allows the algorithm to naturally exploit scene dynamics to improve stereo matching. Methods to test and evaluate the event-based stereo vision algorithms developed during this project are explained. A detailed characterization and evaluation of the algorithm yields important insights into its mechanisms. These mechanisms were found to be very significant and can be generally applied in the field of event-based machine vision.

In Chapter 5, the algorithm developed in the previous chapter is reviewed and discussed in the context of other general principles of brain-inspired computation. An interesting similarity with early work on cooperative stereo networks (Marr and Poggio, 1976) is found, which finally leads to the proposal of a *spiking neural network* for stereo vision. The network employs spiking neurons and operates directly on events provided by an event-based camera. Despite its simple architecture and neural models, the network is found to solve the correspondence problem surprisingly well. Similarly to the stereo algorithm, the network naturally exploits temporal scene dynamics to solve the correspondence problem. The network is then compared to established models of stereopsis and very interesting analogies are found. This even leads to a proposed hypothesis on how motion cues might be integrated in the brain to solve the correspondence problem.

Chapter 6 aims to map the stereo network onto neuromorphic hardware in order to provide an efficient approach to event-based stereo vision. For this purpose, a custom-made, versatile, neuromorphic multi-chip setup comprising various components is described. At the heart of this setup is a neuromorphic processor which was partially designed within the scope of this thesis. Various configurations of the setup are used to explore the neuromorphic circuits which implement components of the stereo network. This yields crucial insights into how neuromorphic hardware could be further developed in order to implement a full-scale stereo network. Finally, different applications of event-based stereo vision are explored and for each application an optimal approach is proposed based on the findings of this thesis. A feasibility study on applying event-based stereo vision to virtual reality is carried out and compellingly demonstrates the potential benefit of event-based stereo vision in a practical, real-world case.

In Chapter 7, the work is summarized and concluded. A dedicated section on future work

describes an *event-based neuromorphic stereo vision system*. This final proposal unifies all of the insights gained from Chapters 4, 5 and 6 within a system that, if properly realized, could revolutionize the field of machine stereo vision.

2 A Review of Stereo Vision

The aim of this chapter is to provide an overview of two largely independent fields that study the same scientific problem: *stereopsis* and *machine stereo vision*. Firstly, the common underlying scientific problem — known as the *stereo correspondence problem* — is introduced, followed by a review of stereopsis. Then, a transition to machine stereo vision is made through a common interface, referred to as *cooperative stereo vision*. Finally, the field of machine stereo vision is reviewed followed by a short discussion of the entire chapter.

2.1 The stereo correspondence problem

Consider a simplified scenario involving a two-dimensional world on a horizontal plane. Such a world can be fully described using two coordinates, the horizontal position and depth. In this world, an observer's eyes project the horizontal position of an object onto one-dimensional retinas, whereby the indices x_R and x_L denote the corresponding positions in the retinal images. This scene is illustrated in Fig. 2.1. A scene object (denoted by the black rectangles) is projected onto each retina if it is within the blue lines of sight. The disparity of an object d is defined as the difference in the positions of its projection onto each of the retinas:

$$d = x_R - x_L \tag{2.1}$$

Here, the two scene objects (black rectangles within the field of view) are said to have zero disparity because they project to the exact same retinal positions on both images. The set of all points with zero disparity is known as the horopter. Points that are located closer to the eyes than those on the horopter are said to have *crossed disparity* ($d < 0$), whereas those which lie further away are referred to as points with *uncrossed disparity* ($d > 0$). In the scientific study of vision, a feature refers to a specific structure in the image itself, ranging from simple structures such as points or edges, to more complex structures such as objects [1]. The aim of stereo

[1] http://en.wikipedia.org/wiki/Feature_(computer_vision)

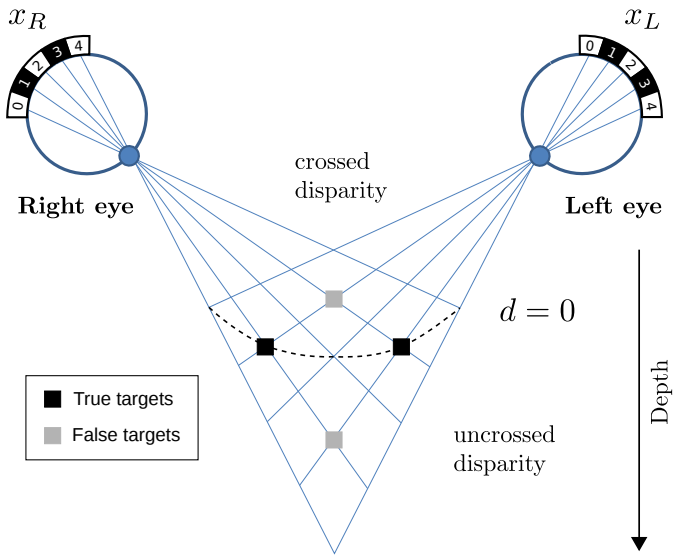


Figure 2.1: The projection of a 2D scene onto the left and right eye. The horopter (dashed line) indicates points with zero disparity. False targets arise from ambiguity among features.

vision is to reconstruct the observed scene from the retinal images. Thus, correspondences among features have to be established. In general, this proves to be a difficult challenge as typical scenes consist of many similar features. This is known as the *stereo correspondence problem*. In the example illustrated in Fig. 2.1, the objects produce similar features at retinal positions 1 and 3, leading to so-called *false targets* (shown in grey) which are features that have been erroneously matched. Like many other problems related to early vision, stereo correspondence is an ill-posed problem and to resolve it, certain assumptions about the scene need to be made.

2.2 The physiology of stereopsis

The idea of binocular cells that merge the input from corresponding retinal regions dates back to Ramón y Cajal (1909) and was confirmed by Hubel and Wiesel (1962) who found such cells in the visual cortex of cats. These cells had receptive fields at corresponding retinal positions, which implies that they could only encode objects with zero disparity. Thus, this did not explain how objects of differing disparities could be simultaneously perceived. In order for that to be possible, a set of binocular cells with varying receptive-field offsets would be required. Such cells were first discovered by Barlow et al. (1967) in the visual cortex of cats and led to the definition of *disparity detectors*. Since then, disparity detectors have been extensively studied and were found to be the most important component of stereopsis. The following review of stereopsis is chiefly informed by the first chapter of an outstanding book on stereoscopic vision by Howard and Rogers (2012) and review papers by Cumming and DeAngelis (2001) and Blake and Wilson (2011), which focus on topics that are particularly

relevant to this thesis.

2.2.1 Brain regions involved in stereopsis

While disparity-tuned cells already exist in some sub-cortical areas, it seems that their selectivity is derived from the visual cortex rather than from the retinas themselves. Disparity-tuned cells have been found in the cat's pulvinar nucleus and the nucleus of the optic tract (NOT), but not in the lateral geniculate nucleus (LGN) (Xue et al., 1987). The LGN is divided into layers of parvocellular and magnocellular cells. The parvocellular neurons respond to color and exhibit higher spatial resolution than the magnocellular neurons. Conversely, magnocellular neurons have higher temporal resolution but are only sensitive to luminance. It is important to note that the parvocellular system is purely chromatic for low spatial and low temporal frequencies. In the case of high spatial and temporal frequencies, however, the parvocellular system shows photometric additivity and conveys pure luminance signals. Through selective lesion of either system, it can be shown that fine stereopsis is confined to the parvocellular system, whereas both systems are capable of detecting low-frequency disparities. A large number of cells in the superior colliculus are sensitive to coarse disparities, suggesting that these cells serve to control *vergence eye movements* or fixation on stimuli that move in depth (Bacon et al., 1998; Mimeault et al., 2004). The visual cortex provides the main input for the superior colliculus, containing a topographic map of visual space (Graybiel, 1976). It is not known whether this map extends to the third dimension. There are two types of disparity detectors, which respond either to position or phase disparity, both of which are located in the superior colliculus. These types are explained in Section 2.2.2. The primary visual cortex (V1) is the first site where disparity selectivity occurs (Cumming and DeAngelis, 2001). The mechanisms underlying disparity detection in V1 are very well understood. Conversely, much less is known about disparity processing in the higher visual areas. In addition to the simple disparity detectors in V1, more specific cells that are also sensitive to relative disparity, depth discontinuities, motion and shape are located in the extrastriate areas V2, V3, V3A, MT and MST. Fig. 2.2 shows the visual pathways from the retina to the cortex in the human brain.

2.2.2 Disparity detection in primary visual cortex

In the visual cortex of primates, neurons that are selective to disparity were first detected in V2 by Hubel and Wiesel (1970) and later in V1 and other visual areas. In V1, more than half of the cells of both physiological types — simple and complex cells — were found to be disparity selective. Experimental studies showed that complex cells had an increased sensitivity to disparity in random-dot stereograms. This suggested that complex cells specifically encode information about the relationship between the two images in the monocular receptive fields. Conversely, simple cells respond to arbitrary excitatory stimuli in their receptive fields and thus, the information about disparity may be disturbed by artifacts of the stimulus shape and location.

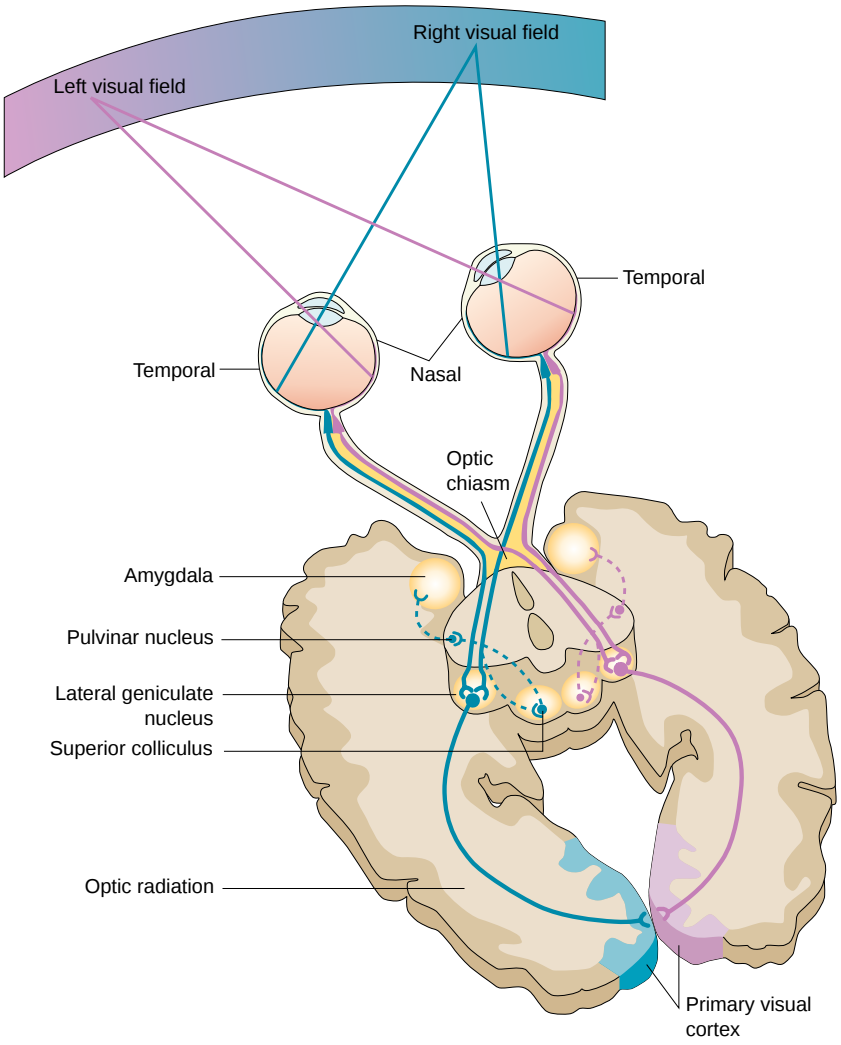


Figure 2.2: The visual pathway from the retina to the cortex. Redrawn from *Nature Neuroscience*.

Characteristics of disparity detectors

Disparity detectors are characterized by their disparity-tuning function, which they perform by encoding the firing frequency depending on the disparity of the retinal images. The width of the tuning function, measured at half its peak amplitude, determines the *selectivity* of a detector. A narrow tuning function denotes high disparity selectivity. The preferred disparity of a detector corresponds to the degree of disparity at which it elicits the strongest response. Poggio (1991) classified disparity detectors into six main classes according to their tuning function: *near* (NE) cells broadly tuned to cross disparities, *far* (FA) cells broadly tuned to uncrossed disparities, *tuned excitatory* (T0) cells with a narrow tuning function near zero disparity, *tuned inhibitory* (TI) cells with a reversed narrow tuning function near zero disparity, *tuned near* (TN) cells with a preference for crossed disparity and *tuned far* (TF) cells with a preference for uncrossed disparity. Prototypical examples of these six types are illustrated

in Fig. 2.3. Typically, T0 and TI cells peak within ±12 arcmin of zero disparity while TN and TF cells have well defined peaks at up to ±0.5 deg. It seems plausible that the near and far cells provide the basis for coarse stereopsis, whereas the narrow disparity detectors are used to discriminate fine disparities. However, this would require many fine-tuned detectors located at different disparities. Alternatively, it is also possible that broad channels encode fine disparity levels similarly to the way in which broad chromatic channels discriminate fine colors. Prince et al. (2002) examined the responses of many disparity-tuned neurons in V1 of an alert monkey and observed no evidence of clustering. They concluded that tuning curves cannot be meaningfully divided into separate types and that they seem to form a continuum instead. Just as the hue-discrimination function contains humps caused by the discrete nature of chromatic channels, the disparity-discrimination function would have been expected to exhibit such humps if a fixed set of discrete channels was used. However, such evidence was not found (Stevenson et al., 1992; Cormack et al., 1993), suggesting that the disparity tuning functions form a continuum. Nevertheless, it should be considered that receptive field sizes are not homogeneous, meaning that the bandwidth of disparity detectors also varies. This could also account for the absence of humps in the disparity-discrimination function. Prince et al. (2000) assessed the precision of single disparity neurons in V1 by comparing the neural response with the behavioral response of alert monkeys to a task involving stereoscopic depth discrimination. They found that the neural response to absolute disparities was more sensitive than the behavioral response, whereas the opposite was the case for relative disparities. This suggests that the behavioral threshold is driven by relative disparity, while the neurons in V1 tend to mostly encode absolute disparity.

Simple cells are characterized by linear spatial summation of inputs within their receptive field. This lead to the definition of *receptive field maps*, which encode the cell's response at each position of the receptive field. Such maps are well described in the existing literature, such as in the functions provided by Gabor filters in the case of simple cells in V1 (Jones and Palmer, 1987). The neural response of simple cells can be predicted by convolving an input image with the receptive field map. When the monocular and binocular receptive field maps of simple cells for sinusoidal-luminance grating stimuli were compared at different disparities, the binocular response was found to be well described by the summation and rectification of the monocular responses. This conforms with the idea of an energy model that computes a binocular contrast energy. The model predicts that disparity detectors of this kind exhibit a phase-specific response, meaning that they are not only sensitive to disparity but also to the stimuli's position within the receptive field (i.e. they are not position invariant). While this was experimentally confirmed for simple cells, however, the disparity responses of complex cells tend to show position invariance (Hammond, 1991). This observation led to the development of the *famous disparity energy model* (Ohzawa et al., 1990). The model explains how the neural response of complex binocular cells is computed from the responses of multiple simple binocular cells in order to obtain position invariance. It was suggested that these neurons are optimally suited to be disparity detectors. To accord with the energy model, disparity detectors should exhibit an inverted tuning function if the sign of the contrast

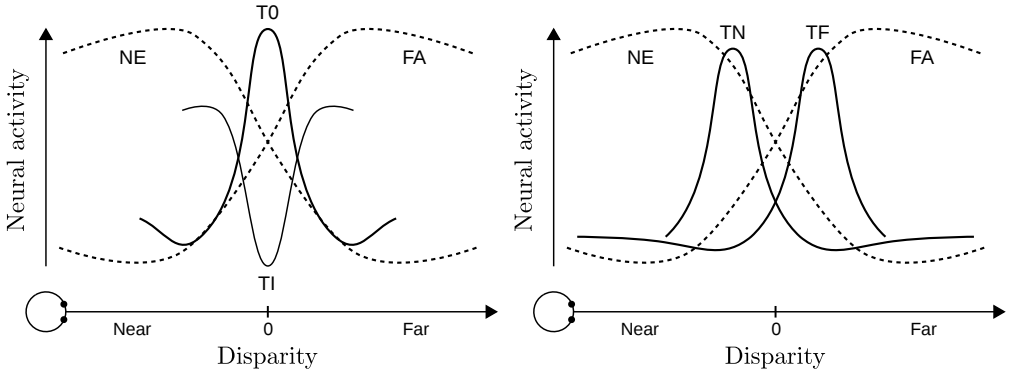


Figure 2.3: Ideal types of disparity tuning functions. The tuning functions of broad near (NE) and far (FA) cells are shown for reference in both plots. Tuned excitatory (T0) and inhibitory (TI) functions are shown on the left, whereas tuned near (TN) and far (TF) functions are shown on the right. Redrawn from Poggio et al. (1985a).

is reversed in one eye. Cumming and Parker (1997) demonstrated that disparity selective neurons in V1 were able to detect disparity in anti-correlated stereograms (in which matching dots have opposite contrast) while it had been shown earlier that depth is not perceived in such stereograms (Cogan et al., 1993). This led to the conclusion that V1 neurons do not unambiguously signal depth. The disparity energy model is further discussed in Section 2.2.5.

Primary disparity detectors are cells that receive their input directly from the retinas. They occur in V1 and encode *absolute* local disparities. In contrast, secondary disparity detectors combine the signals from primary detectors and respond to *relative* disparities. Usually, secondary detectors and similar mechanisms of higher order are found in the extrastriate cortex. Cumming and Parker (1999) studied V1 neurons in alert monkeys and found that while most of the observed neurons responded to absolute disparities, not a single neuron maintained a consistent relationship with relative disparity. This suggests that the disparity detectors in V1 are essentially of the primary type. In a subsequent study, Cumming and Parker (2000) found that V1 neurons signaled zero disparity of stimuli that did indeed have local ambiguous evidence of zero disparity, but that they were correctly perceived to be at a depth distant from the fixation point. This supports the idea that neurons in V1 are primary detectors which encode absolute local disparity. However, conflicting evidence of the existence of secondary disparity detectors in V1 has been recently published by Sasaki et al. (2010).

Position and phase mechanism

Two different mechanisms exist to detect disparity. At first, only *position disparity* was considered relevant, which corresponds to the case where a binocular cell receives its input from two monocular receptive fields that are identical in shape and only differ in horizontal retinal position. The monocular receptive fields are often modeled using Gabor functions

$$\begin{aligned} RF_L &= \exp\left(\frac{-x^2}{\sigma^2}\right)\cdot\cos(\omega x) \\ RF_R &= \exp\left(\frac{-(x-d)^2}{\sigma^2}\right)\cdot\cos\left(\omega(x-d)\right) \end{aligned} \tag{2.2}$$

where x is the retinal position, σ the width of the Gaussian envelope, ω the preferred spatial frequency and d the disparity. In contrast, *phase-disparity* detectors employ receptive fields located at the same retinal positions, but which differ in phase:

$$\begin{aligned} RF_L &= \exp\left(\frac{-x^2}{\sigma^2}\right)\cdot\cos(\omega x) \\ RF_R &= \exp\left(\frac{-x^2}{\sigma^2}\right)\cdot\cos\left(\omega x+\phi\right) \end{aligned} \tag{2.3}$$

The associated receptive fields for both mechanisms of disparity detection are illustrated in Fig. 2.4. Evidence of phase detectors was found in the visual cortex of the cat (Freeman and Ohzawa, 1990; DeAngelis et al., 1991) and monkey (Livingstone and Tsao, 1999), which has been further supported by psychophysical experiments on human subjects (Edwards and Schor, 1999). Position-disparity can be arbitrary and thus, there is not necessarily a relationship between the size of the receptive field and the preferred disparity. For example, a binocular cell could have small receptive fields with a large interocular offset, meaning that it could detect large disparities. Conversely, in the case of phase-disparity detectors, there is an immutable link between the size of the receptive field and the preferred disparity of the detector due to the periodicity of the receptive field. The maximum detectable disparity of phase detectors is limited to half a cycle of the spatial period and correct matches are only possible within a quarter-cycle of the spatial period (Howard and Rogers, 2012). This implies that simple cells tuned to high spatial frequencies with small receptive fields are restricted to the detection of small disparities. However, considerably larger disparities for high spatial frequencies have been detected than would be predicted based on the restrictions of the phase detectors (Schor et al., 1984). This suggests that the visual system combines both position and phase-disparity mechanisms. Several explanations for the existence of two distinct types of disparity detectors exist. The work of Qian and Andersen (1997) has revealed that when receptive fields are modeled using Gabor functions, populations of phase-disparity detectors can yield sharper responses. It concludes that phase-disparity detectors are used to provide more accurate disparity responses. Read and Cumming (2007) propose an alternative interpretation of the purpose of phase detectors. The authors argue that the optimal stimuli for phase-disparity detectors do not occur in a real scenario because of the different shapes of the receptive fields that would respond optimally to non-identical stimuli. They contend that phase detectors respond more strongly to correspondences if the image features are similar

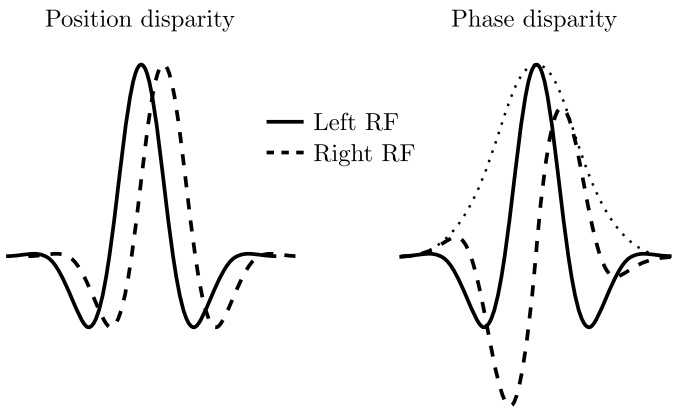


Figure 2.4: Mechanisms of position and phase disparity detection. Redrawn from Cumming and DeAngelis (2001).

but not identical. Thus, the authors posit that their role is to detect false matches.

Stereo correspondence in V1

It is the subject of ongoing debate whether neurons in V1 are capable of solving the stereo correspondence problem. While simple cells are not well suited to be disparity detectors because they do not show position invariance, it may be possible that complex cells are sophisticated enough to compute unambiguous disparity. In order to investigate this, complex cells should be tested with stimuli that produce ambiguous disparities. Random-dot stereograms (RDS) contain such ambiguity but the responses are often averaged over several trials, whereby the ambiguity produced by the uncorrelated dots changes from trial to trial, and thus, cancels itself out. In this case, even the simplest model responds maximally to correct matches, as pointed out by various researchers (Fleet et al., 1996; Cumming and DeAngelis, 2001). The traditional energy model cannot resolve ambiguity. This has been demonstrated by the response of complex cells to anti-correlated RDS, which signaled disparity that was not perceived as depth (Ohzawa et al., 1990; Livingstone and Tsao, 1999). However, the linear pooling of responses across spatial frequency and orientation could produce an unambiguous representation (Fleet et al., 1996). These observations suggest that complex cells perform an initial step of binocular correlation that is later used in higher visual areas to solve the correspondence problem (Cumming and DeAngelis, 2001). Additional evidence supporting the hypothesis that further processing is required beyond the striate cortex was obtained by Cumming and Parker (2000), which showed that V1 neurons signal absolute local disparities in ambiguous stimuli, irrespective of the perceived depth. Despite this overwhelming evidence, however, there are reasons to believe that complex cells are more sophisticated and play a more important role in discriminating between correct and false disparities. One contradiction with the energy model is that disparity tuning curves for anti-correlated RDS are often attenuated (Ohzawa et al., 1997). Another reason is provided by the role of phase-disparity detectors, which are believed to signal false matches and feed into complex cells as suppressive components to

reduce responses to ambiguity (Read and Cumming, 2007; Tanabe et al., 2011).

2.2.3 Disparity detection in higher visual areas

Binocular cells occur in V2 more frequently than in V1, in which a considerable proportion of cells seem to be secondary disparity detectors. Thomas et al. (2002) found neurons in V2 of an alert monkey, that showed selectivity for relative disparities. They proposed a model which involved summing the outputs of two neighboring primary disparity detectors followed by nonlinear half-squaring. It was suggested that such detectors are crucial to discriminate psychophysical depth. Cells that are sensitive to the sign of depth at an edge were found by Qiu and von der Heydt (2005). The authors suggested that disparity detectors are coupled with cells that are selective to boarder ownership of 2D figures, and therefore, play a crucial role in the figure-ground articulation of the brain. Conversely, Bredfeldt and Cumming (2006) proposed that cells tuned to disparity edges simply arise from the integration of several responses from cells in V1 with different disparity selectivity.

Disparity detectors also occur in the middle temporal (MT) and medial superior temporal (MST) area, which are part of the dorsal stream. The dorsal stream is associated with visually guided behavior and involves the coding of spatial location, motion and coarse stereopsis. Generally, the same type of disparity detectors found in V1 also occur in MT but they tend to be more broadly tuned. Recordings from near and far cells of an alert monkey have been shown to predict the animal's perception of whether an object is nearer or beyond a fixation point (Uka and DeAngelis, 2004). Based on experiments involving microstimulation, Uka and DeAngelis (2006) proposed that MT neurons encode an indication of disparity but not fine relative disparities. Cells that are jointly tuned to motion and disparity have been found in MT as well as MST. Dodd et al. (2001) showed that cells in MT capable of signaling the direction of motion at different depths predict a monkey's perception of the direction in which a 3D stimulus is rotating. They used a random-dot transparent cylinder rotating around a horizontal axis. The direction of rotation was ambiguous if the cylinder's diameter was set to zero, which revealed the correlation between alternating MT responses (due to ambiguity) and the perception of a monkey. The observed link between MT neurons and psychophysical perception leads to the question of whether these neurons also signal disparity that is not perceived as depth — as is the case for anti-correlated RDS, for example. Indeed, Krug et al. (2004) have found neurons in MT that are insensitive to disparity in anti-correlated RDS, but also neurons that signaled disparity. In addition, cells in MST were found to be selective for disparity in anti-correlated RDS (Takemura et al., 2001). On this basis, it is not clear whether single MT neurons encode disparity unambiguously. However, it is known that MT and MST are associated with the control of vergence and it is believed that anti-correlated disparities serve as a trigger for vergence eye movement. Further disparity selective neurons along the dorsal path are found in the parietal cortex. There, neurons that respond to the orientation of 3D objects were found.

The ventral visual pathway is the stream that is associated with pattern recognition and fine stereopsis. Hinkle and Connor (2005) found that the majority of V4 neurons in the macaque are tuned to fine disparities in the range of $-1°$ to $+1°$. The tuning functions were found to be consistently distributed between the classical types of tuned excitatory, tuned inhibitory and near and far functions. Hinkle and Connor (2002) recorded V4 neuron responses that were shown to specifically encode the slant of bars that were tilted in depth. It was suggested that the signals tuned to 3D orientation are used to recognize objects higher up in the ventral pathway. The inferior temporal cortex (TE) accommodates cells that respond to the 3D orientation of textured surfaces, irrespective of the texture itself (Liu et al., 2004). The majority of cells in TE are disparity detectors of higher order, that respond to disparity along a contour of a curved surface, or to disparity gradients within the surface. Some other cells were tuned to 3D shapes characterized by disparity (Tanaka et al., 2001). Shape-selective neurons in TE showed robust selectivity for correlated RDS, but were not selective for anti-correlated RDS, which suggested that at the level of the inferior temporal cortex, the stereo correspondence problem seems to be solved (Janssen et al., 2003).

2.2.4 Joint encoding of disparity and motion

Cells in higher visual areas exhibit patterns of relative disparity, whereby a relationship exists between either disparity and motion, or spatial and temporal disparity. Anzai et al. (2001) proposed that a common neural mechanism may be used to encode motion and disparity, as they are both defined by the difference in an object's position either across the eyes, or over time. They found that most cells showed a space-time-oriented response, suggesting that motion and disparity are jointly encoded at an early visual stage, which could explain certain illusions such as the Pulfrich effect. Pack et al. (2003) found cells that showed modest tuning to both spatial and temporal disparity in V1 and MT of alert monkeys. The authors mapped the spatio-temporal receptive field of these cells and observed a slant which represented the cell's selectivity for a specific combination of spatial and temporal disparities. In fact, these cells cannot distinguish between a real positional disparity and a zero disparity, if it is presented with an interocular temporal delay. It was suggested that these cells are involved in the process of motion parallax, and can explain psychophysical evidence such as the Pulfrich effect.

Motion in depth

Much of the recent work on the relationship between motion and disparity deals with motion in depth. Motion in depth can be computed from two cues (Regan, 1993). The first cue is the difference in inter-ocular velocity between two monocular motion signals. The second cue is the rate at which disparity changes relative to a fixed object. Cumming and Parker (1994) measured the thresholds for stereomotion with stimuli that only contained one of the two cues. They found that perception of stereomotion is mainly based on the rate at which disparity changes, whereas the difference in inter-ocular velocity plays only a minor role. Conflicting evidence was provided by Shioiri et al. (2000), which showed that a discrimination

task involving stereomotion can be successfully solved based on binocular uncorrelated random-dot kinematograms that solely contain a difference in inter-ocular velocity, but no disparity information. Functional magnetic resonance imaging (fMRI) studies with human subjects revealed that motion-in-depth processing is associated with the human motion complex (MT+) (Likova and Tyler, 2007; Rokers et al., 2009).

2.2.5 Neural models of stereopsis

Two kinds of computational models of stereopsis can be distinguished: energy models and neural networks. Energy models are based on the idea that binocular neurons compute the disparity energy from monocular receptive fields that are either shifted in position or phase. They do not account for global features, such as the interactions of disparity neurons; instead, only local disparities within their receptive field are considered. These models were developed to precisely predict the responses of disparity-tuned cells and are based on the knowledge about simple and complex cells in V1. On the other hand, neural networks can explain various aspects of stereopsis including global features, but are only loosely linked to real biological mechanisms.

Disparity energy models

The earliest cortical binocular neurons are simple cells that occur in V1 of primates. Simple cells are characterized by linear summation of the monocular receptive fields followed by a threshold nonlinearity. The monocular receptive fields were found to be well described by elongated Gabor functions, which consist of a sinusoidal carrier modulated by a Gaussian envelope:

$$\begin{aligned} RF_L &= \frac{1}{\sqrt{2\pi\sigma_{lx}\sigma_{ly}}}\exp\left(-\frac{(x-x_l)^2}{2\sigma_{lx}^2}-\frac{(y-y_l)^2}{2\sigma_{lx}^2}\right)\cdot\cos\left(\omega_{lx}(x-x_l)+\phi_{lx}\right) \\ RF_R &= \frac{1}{\sqrt{2\pi\sigma_{rx}\sigma_{ry}}}\exp\left(-\frac{(x-x_r)^2}{2\sigma_{rx}^2}-\frac{(y-y_r)^2}{2\sigma_{rx}^2}\right)\cdot\cos\left(\omega_{rx}(x-x_r)+\phi_{rx}\right) \end{aligned} \tag{2.4}$$

The preferred spatial frequency of the receptive fields are defined by ω and the shift in phase by ϕ. The standard deviations of the Gaussian envelope are denoted with σ. Here, the receptive fields have a preferred orientation which is aligned with the vertical axis (y axis) but in general any arbitrary orientation could be obtained by simply rotating the profiles in Eqs. 2.4. Usually, it is assumed that both monocular receptive fields have the same orientation. If the carrier is in cosine phase, the receptive field is called *even-symmetric* while an *odd-symmetric* field would refer to a carrier which is in sine phase. A binocular simple cell responds most strongly to a luminance gradient with a periodicity that matches the preferred spatial frequency and is aligned with its preferred orientation. The basic disparity energy model only considers

disparities that are aligned with the preferred orientation of the simple cell. Therefore, we can reduce Eqs. 2.4 to the one-dimensional case and assume that both monocular receptive fields have an equal width σ and equal spatial frequency ω:

$$\begin{aligned} RF_L &= \frac{1}{\sqrt{2\pi\sigma}}\exp\left(-\frac{(x-x_l)^2}{2\sigma^2}\right)\cdot\cos\left(\omega(x-x_l)+\phi_l\right) \\ RF_R &= \frac{1}{\sqrt{2\pi\sigma}}\exp\left(-\frac{(x-x_r)^2}{2\sigma^2}\right)\cdot\cos\left(\omega(x-x_r)+\phi_r\right) \end{aligned} \tag{2.5}$$

These receptive fields act as linear filters, whereby the cell's response can be obtained by convolving the receptive field map with the stimulus function. The response r reflects the firing rate of the cell. A binocular simple cell combines the inputs from two eyes and its response can be approximated by

$$r_s = \int_{w/2}^{w/2} \left[RF_L(x)I_L(x) + RF_R(x)I_R(x)\right]dx \tag{2.6}$$

where w is the width of the receptive field and $I_L(x)$ and $I_R(x)$ correspond to the stimulus as seen in the left and right retina respectively. The responses of binocular simple cells are half-wave rectified because their monocular receptive fields only respond to one type of polarity, either an increase in light (ON-type) or a decrease in light (OFF-type). Anzai et al. (1999) have found that the response of binocular simple cells is better predicted when a static nonlinearity in the form of a power function is added. The average exponent was found to be close to 2, which corresponds to a squaring mechanism. The authors pointed out that this form of multiplicative interaction could be an important step in computing interocular cross-correlation and solving the correspondence problem. It has been previously mentioned that binocular simple cells show position sensitivity. This can be seen when solving Eq. 2.6 for a stimulus $I(x)$ with binocular disparity such that $I_L(x) - I(x)$ and $I_R(x) = I(x+D)$. Assuming that $w \gg D$, the Gaussian envelope can be ignored, resulting in the following expression for the response (Qian, 1994):

$$r_s \approx 2|I(\omega)|\cos\left(\angle I(w) + \frac{\phi_l+\phi_r}{2} - \frac{\omega D}{2}\right)\cos\left(\frac{\phi_l-\phi_r}{2} + \frac{\omega D}{2}\right) \tag{2.7}$$

Here, $|I(\omega)|$ and $\angle I(\omega)$ are the magnitude and phase of the Fourier transform of the stimulus at the preferred spatial frequency. The Fourier phase of the input stimuli is only invariant when the baseline shift or scaling of brightness is constant (Qian, 1994), which explains why the simple cell is dependent on the stimulus position and sign of contrast, as observed by Ohzawa et al. (1990). Qian (1994) has also shown how the tuning behavior of a simple cell can change from tuned excitatory to the tuned near and far types, simply by changing the parameters that

affect the Fourier phase.

In Section 2.2.2, the mechanisms of position and phase-disparity were introduced. Position-disparity detectors are obtained when the receptive fields from Eqs. 2.5 are identical except for a displacement in retinal position. The preferred disparity is then directly coded according to the difference in the offset between the left and right monocular receptive fields, i.e. $d = x_r - x_l$. Conversely, phase-disparity detectors are obtained when position offsets are equal, $x_r = x_l = x_0$, but the phase shifts ϕ_l and ϕ_r vary. In this case, the disparity depends on the spatial frequency. If a stimulus is presented that has a spatial frequency which matches the cell's preferred frequency ω, the following relationship between position and phase disparity can be derived:

$$d = \frac{\phi_r - \phi_l}{\omega} \tag{2.8}$$

The general definition of the receptive fields in Eqs. 2.4 allows the construction of further disparity detectors. If a binocular cell combines the inputs from two monocular receptive fields with varying preferred spatial frequencies, it will be sensitive to disparities at slanted surfaces. Alternatively, binocular cells that combine receptive fields with varying preferred orientations are tuned to shear disparities, as they occur at inclined surfaces. The problem with energy neurons is that they not only respond to the disparity energy, but also to the monocular energy from each receptive field, which leads to distortion of the disparity signal. To account for the contribution of the monocular components, Hibbard (2008) introduced a divisive normalization procedure. However, by comparing the response of neurons at the same location with diverse preferred disparities, an estimate of the monocular energy components can be yielded. Either this, or the response of monocular cells themselves, can be used to decode the disparity response of energy neurons.

Unlike simple cells, binocular complex cells exhibit position invariance, which is achieved by summing the responses of quadrature pairs of simple cells (Ohzawa et al., 1990; Qian, 1994; Anzai et al., 1999). A quadrature pair of cells has receptive fields that differ in phase by 90°. The response of a binocular complex cell can thus be written as

$$r_c = r_{s1}^2 + r_{s2}^2 \tag{2.9}$$

which can be solved using the result from Eq. 2.7 and choosing the phase offsets such that $\phi_{l1} - \phi_{l2} = \phi_{r1} - \phi_{r2} = \pi/2$. The resulting response no longer depends on the Fourier phase:

$$r_c \approx 4|I(w)|^2 \cos^2\left(\frac{\phi_l - \phi_r}{2} + \frac{\omega D}{2}\right) \tag{2.10}$$

The final model is schematically illustrated in Fig. 2.5. In order to obtain a disparity response

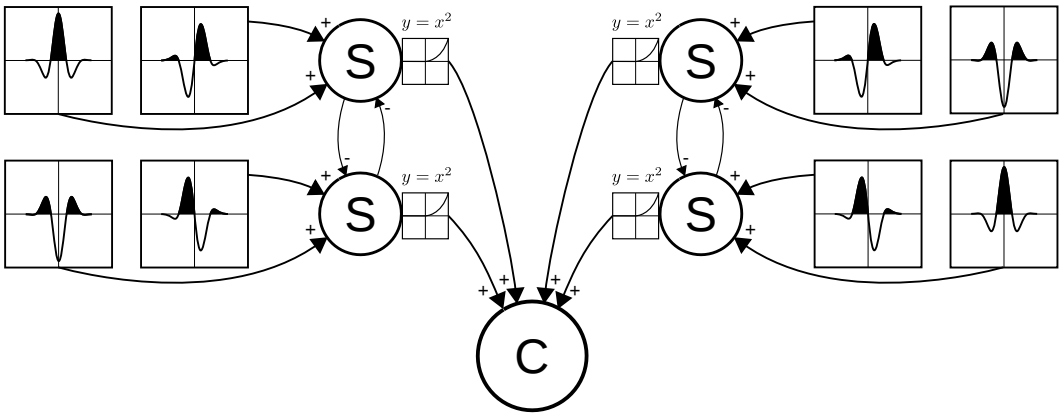


Figure 2.5: Disparity energy model of binocular simple and complex cells. Simple cells (S) linearly add a pair of monocular receptive fields during the quadrature phase, followed by a half-wave rectified squaring. Selectivity to both polarities is modeled with two mutually inhibitory cells featuring reversed receptive field profiles which form a subunit. The complex cell (C) receives input from at least two subunits. Redrawn from Ohzawa et al. (1990).

that is independent from the polarity of the stimulus, a combination of two similar quadrature pairs, one with ON-center and one with OFF-center cells, is used. When the disparity energy model is applied to an RDS, a sort of smoothing operation is often required to get rid of false matches (high frequencies). Such operations simply involve pooling several responses (Qian, 1994) or weighted averaging of responses from several quadrature pairs of simple cells (Qian and Zhu, 1997). The latter approach has the advantage that it preserves disparity discontinuities. In the model of Fleet et al. (1996), a local disparity energy function is obtained by pooling several responses across scale, orientation and local space. Disparity detectors tuned to a certain disparity show a peak in the energy function at the preferred disparity, if it corresponds to a true disparity. The exact disparity at a given spatial position could then be estimated by interpolating the responses from a discrete set of cells tuned to different disparities at the corresponding spatial position. Based on physiological evidence, Prince and Eagle (2000) introduced a bias in favor of small disparities into the energy functions in order to reduce the responsiveness to false matches.

Interestingly, there is biological evidence that challenges the standard disparity energy model of complex cells. Livingstone and Tsao (1999) have pointed out that simple cells are rare in the macaque monkey and have found no evidence that complex cells receive their input from disparity-tuned simple cells. Furthermore, some complex cells in the cat seem to receive their input directly from LGN neurons (Alonso and Martinez, 1998). This led to the proposal of a related energy model, whereby the computation of preceding subunits that feed into complex cells is not performed by simple cells, but directly by branches of active dendrites (Alonso and Martinez, 1998). Read and Cumming (2007) and Haefner and Cumming (2008) have shown that energy neurons with pure phase disparity respond optimally to compound gratings with sinusoidal components displaced by constant phase. However, such stimuli appear differently shaped to the left and right eye, thus suggesting that phase-detectors are most selective for

stimuli that do not occur in the real world. Solid surfaces cannot produce constant interocular phase differences in all Fourier components. Read and Cumming (2007) showed that neither energy neurons with pure position-disparity, nor those with pure phase-disparity can signal the true disparity of even simple real-world stimuli. The reason is that energy neurons not only respond to the cross-correlation of interocular images, but also to the local contrast. This means that a false match (low interocular cross-correlation) could still produce a higher energy than a true match, if the local contrast is considerably higher. However, a combination of phase and position-disparity detectors could solve this problem, based on observations which show that false matches are likely to contain more phase-disparity, whereas true matches must contain zero phase-disparity. Therefore, it has been suggested that phase-detectors play a crucial role in signaling false targets and might therefore be added as suppressive elements in the standard energy model (Read and Cumming, 2007). Haefner and Cumming (2008) have introduced a generalized disparity energy model, which posits that complex cells combine the output from simple cells of differing disparity selectivity, leading to so-called adapted energy neurons. These cells explain two important features of the visual system that cannot be explained with the standard energy model. Firstly, adapted neurons are tuned to real-world stimuli, which explains why the visual system is also sensitive to them. Secondly, adapted neurons explain the attenuated response to anti-correlated RDS discussed earlier in this chapter.

From disparity energy to depth perception

The vast bulk of evidence suggests that neurons in V1 are selective to absolute disparity (which is well described by the energy model), but that the stereo correspondence problem — which is related to depth perception — is solved beyond the striate cortex. Several models propose that depth perception is computed from a population response of energy neurons across different scales (spatial frequencies). This could be simply achieved by pooling the response of multiple neurons Qian (1994). Alternatively, the population response could also be preserved and compared to responses of white-noise stimuli at different depths, whereby the best match indicates the perceived depth (Tsai and Victor, 2003). This model was found to explain many psychophysical observations such as the dependence of stereo acuity on spatial frequency.

It is commonly assumed that the process of stereopsis is based on local cross-correlation of retinal image patches. This implies that disparity detectors are tuned to constant disparity in locally fronto-parallel surfaces, a property that has been observed in disparity selective neurons (Nienborg et al., 2004). Energy neurons perform a form of cross-correlation, but it is not clear how they handle normalization. For example, it would be possible that energy neurons compute covariance, whereas normalization is carried out at a later stage of processing. However, the concept of cross-correlation is interesting, not least because it explains some crucial aspect of stereopsis. Banks et al. (2004) and Filippini and Banks (2009) have shown that a stereo matching process based on local cross-correlation can explain the well-known constraints of stereopsis. The first constraint is the disparity-gradient limit, which refers to

the inability to perceive depth at locations where the change of disparity exceeds a certain threshold. The second constraint concerns the limits of stereoresolution which, in fact, is much poorer than luminance resolution in humans.

Neural networks

The process of stereopsis can also be modeled with neural networks. These neural networks often rely on abstractions and models that deviate from what is known about real neurons. Becker and Hinton (1992) have shown how artificial neurons can learn to discover depth in RDS of curved surfaces, in an unsupervised fashion. Pairs of interocular modules were trained with a backpropagation algorithm to respond maximally to mutual information contained in the stimuli presented. An additional layer was used to develop a depth interpolation function from the output of the modules, enabling the network to not only respond to frontal but also to slanted and curved surfaces. An interesting aspect of this work is that it eliminates the need for a teacher, because the pairs of modules can generate their own teaching signals by maximizing agreement. Lippert et al. (2000) used monocular Gabor filters with varying interocular phase and position to provide an input to a neural network that was trained with backpropagation to become disparity selective. They showed that the model neurons could successfully learn phase and position-disparity tuning functions that are similar to those predicted by the energy model. However, their model neurons did not explicitly implement the energy model and thus, showed some discrepancies such as significantly reduced selectivity to spatial frequency.

2.3 Cooperative stereo vision: Where neuroscience meets machine vision

The pioneering work of Julesz (1960) suggested that stereo vision is subject to a cooperative process. This hypothesis started a long-lasting discussion about the underlying key principles and neural mechanisms of binocular vision, which led to a variety of theories and algorithms being proposed. The first kind of cooperative algorithms were proposed in the mid 1970's (Dev, 1974; Marr and Poggio, 1976, 1979), at a time when vision was an interdisciplinary field encompassing physiology, psychology, neuroscience and computer science, whereas machine vision did not yet constitute a distinct research area. The algorithms which stemmed from this field, particularly the algorithm famously proposed by Marr and Poggio (1976), provide the main source of inspiration for this thesis, and are described in more detail in the sections which follow.

2.3.1 Cooperative computation of stereo disparity

Studying the physical constraints of the environment, Marr and Poggio (1976) made observations from which the following two general rules were derived:

1. Uniqueness: Each point in each image corresponds to at most one target in the field of view.

2. Continuity: Depth varies smoothly almost everywhere.

The first rule is a direct consequence of the fact that a feature cannot be assigned to multiple objects, as they would occlude each other from the observer's view. The second rule is derived from the cohesiveness of matter. A scene consists of objects which are consistent, causing a smooth variation of depth. Inconsistencies (such as edges) can only be produced by transitions from one object to another and are assumed to occur less frequently. A simple algorithm representing a network that operates on binary images has been proposed to solve the stereo correspondence problem. Fig. 2.6 depicts the behavior of the network. Following the same line of our example in Fig. 2.1, a two-dimensional field of view is considered. The inputs of the network are the one dimensional binary images shown at the top. For each combination of pixel positions x_L and x_R from the left and right retinal image, a cell is placed, each of which represents a point in space corresponding to the intersection of the lines of sight of its associated pixels. A unique disparity value $d = x_R - x_L$ is assigned to each cell. As the cell is active, it reports a true target at its associated position. The resulting network can be thought of as a way of sampling the field of view. The initial state of the network is obtained by setting the units active if both of its associated pixel inputs correlate. Accordingly, the initial state represents the set of all true and false targets. Connectivity among the units is derived from the two rules stated above. The *uniqueness* rule is implemented by inhibition along lines of sight (dashed blue) whereas the *continuity* rule is obtained by excitation along lines of constant disparity (solid red). In the example, it is immediately apparent how the network solves the correspondence problem by converging to the solution — only the true targets are active because they excite each other and inhibit the false targets. While the algorithm works flawlessly on scenes with depths that run parallel to the view of the observer, it fails on surfaces that are tilted in depth. The reason for this shortcoming is that in the case of a tilted surface, the units which are initially active do not lie on the same line of disparity and thus, they cannot excite each other.

2.3.2 Impact of cooperative processes in stereo vision

The work of Marr and Poggio (1976) has inspired many models and algorithms of stereopsis as well as machine vision. On the one hand, as cooperative processes are well known to occur in biological vision systems, they provide a good prospective explanation for how stereo ambiguity could be resolved. On the other hand, while computer algorithms based on cooperative processes approximate a global optimization technique, they have the potential to be more efficient because they are based on cheap iterative local updates. Early cooperative algorithms were not solely concerned with providing a solution to the mathematically ill-posed correspondence problem; they also sought to explain aspects of stereopsis in humans such as the representation and coding of disparity in the brain, the disparity gradient limit,

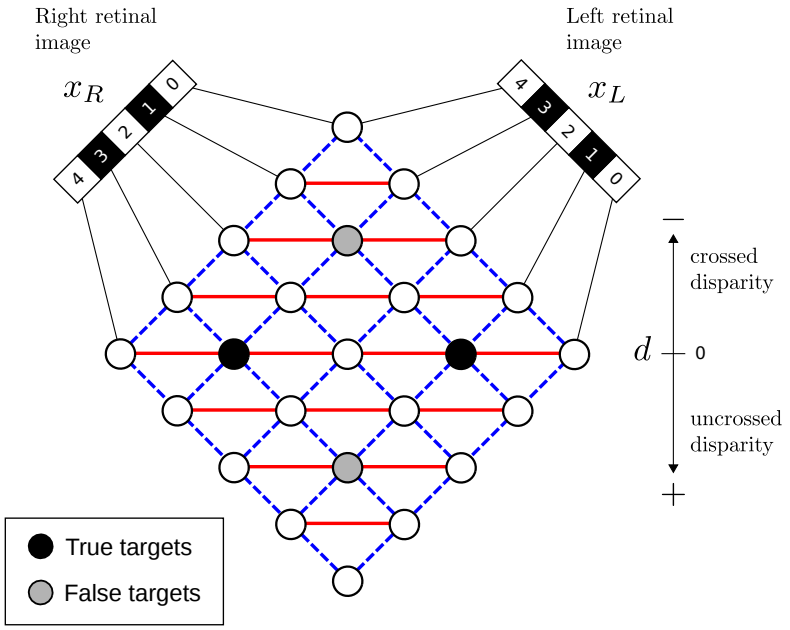


Figure 2.6: A cooperative network for stereo correspondence proposed by Marr and Poggio (1976)

and vergence eye movement (Marroquin, 1983; Prazdny, 1985; Drumheller and Poggio, 1986; Pollard et al., 1986; Mahowald, 1994a). More recent algorithms are still based on principles of cooperative process but have moved away from the analogy of biological systems and no longer try to confirm models or make predictions in the field of stereopsis (Zitnick and Kanade, 2000; Wang and Zheng, 2008). Very recently, cooperative processes have been rediscovered in the field of event-based stereo vision (this work will be discussed in Section 3.5.1), an interdisciplinary research field at the interface of neuroscience and computer science. This provides a very interesting framework for solving the stereo correspondence problem based on new paradigms of computing inspired by the functioning of the brain.

2.4 Machine stereo vision

In this chapter, the essential methods and breakthrough developments in the field of machine stereo vision are summarized. This review is by no means exhaustive, focusing chiefly on approaches relevant to this thesis. In fact, the review mainly rests upon the excellent work of Scharstein and Szeliski (2002). Although the field has been very active in the last decade, many new approaches are based on key methods which had been previously introduced. A detailed review of state-of-the-art approaches is far beyond the scope of this thesis and not relevant to the work in hand. Thus, only the main methods are discussed here, followed by a brief overview of state-of-the-art approaches.

2.4.1 Taxonomy of machine stereo vision algorithms

Scharstein and Szeliski (2002) developed a taxonomy and categorization scheme for machine stereo vision algorithms that has been widely adopted in the literature. They observed that stereo vision algorithms usually consist of subsets of the following four steps (Scharstein and Szeliski, 1998; Scharstein, 1999):

1. Computing the matching cost
2. Aggregating the cost
3. Computing and optimizing disparity
4. Refining disparity

Based on these four steps, stereo vision algorithms can be categorized into three broad classes: *local*, *global* and *iterative* algorithms. The classification is mainly determined by the sequence of the steps. Local, also referred to as *area-based* or *window-based* algorithms, usually compute the matching cost pixel-by-pixel based on intensity values and aggregate this cost over a pixel area with fixed disparity. The disparities with the minimal cost are subsequently selected for each pixel. In contrast, global algorithms seek to assign an optimal disparity that minimizes the global cost. The last class is at the interface of local and global algorithms. Iterative algorithms do not solve an explicit global optimization problem but iteratively improve disparity by repeatedly applying local methods and using the results of previous steps to impose constraints on consecutive disparity searches.

2.4.2 Local algorithms

Local stereo algorithms find matches by aggregating the matching cost of local image patches. Mathematically, this can be expressed as follows: Let $C(x, y, d)$ be the aggregated cost function for each potential match (x, y, d) in the disparity space. Then, a local algorithm can compute the disparity map $D(x, y)$ for each pixel individually as

$$D(x, y) = \underset{d}{\operatorname{argmin}} \big(C(x, y, d)\big) \tag{2.11}$$

which is known as a simple, winner-takes-all (WTA) method. Typical cost functions involve the sum-of-absolute-differences (SAD), the sum-of-squared-differences (SSD) or the normalized-cross-correlation (NCC) (Hannah, 1974):

$$SAD(x,y,d) = \sum_{x,y\in\Omega} \left| i_L(x,y) - i_R(x-d,y) \right| \tag{2.12}$$

$$SSD(x,y,d) = \sum_{x,y\in\Omega} \left(i_L(x,y) - i_R(x-d,y) \right)^2 \tag{2.13}$$

$$NCC(x,y,d) = \frac{\sum_{x,y\in\Omega} \left(i_L(x,y) - \bar{i}_L \right) \left(i_R(x-d,y) - \bar{i}_R \right)}{\sqrt{\sum_{x,y\in\Omega} \left(i_L(x,y) - \bar{i}_L \right)^2 \sum_{x,y\in\Omega} \left(i_R^2(x,y) - \bar{i}_R \right)^2}} \tag{2.14}$$

with Ω being the support region, i_L and i_R the left and right image intensities and $\bar{i}_L$ and $\bar{i}_R$ their associated means in Ω. Unlike the SAD and SDD, the NCC has the advantage that it accounts for variation in intensity gain and offset, at the cost of being computationally more expensive. A comparison and evaluation of different cost functions can be found in Scharstein and Szeliski (2002), Mayoral et al. (2006), and Hirschmuller and Scharstein (2007). In order to avoid redundant computation when aggregating cost, implementation tricks such as *box filters* or *binomial filters* are often used (McDonnell, 1981; Kanade et al., 1996). A box filter is a separable moving average filter in the cost function space, whereas binomial filters are based on separable finite-impulse-response (FIR) filters. Very recent approaches have adopted edge-preserving bilateral filters. These *guided filters* have been shown to achieve a high level of accuracy and real-time performance (Yoon and Kweon, 2006; Rhemann et al., 2011). Another class of local algorithms are feature-based methods that compute a matching cost based on image features which often (and unavoidably) yields only sparse disparity maps (Arnold, 1983). For example, these algorithms calculate a very simple binary matching cost based on the presence of edges (Canny, 1986) or the sign of the Laplacian (Nishihara, 1984). More informative primitives or features include the orientation of gradients (Seitz, 1989), metrics based on gradient-fields (Scharstein, 1994) or phase and filter-bank responses (Kass, 1988; Jenkin et al., 1991; Jones and Malik, 1992). Zabih and Woodfill (1994) have introduced two non-parametric local transformations, that rely on the relative ordering of intensity values, rather than the intensity values themselves. The *rank* transform is defined as the number of pixels in a local region which have a smaller intensity than the central pixel. The *census* transform corresponds to a bit string representing all neighboring pixels whose intensity is less than that of the central pixel. It has been shown that such image transforms are similar to the NCC, invariant to differences in camera gain but less sensitive to image noise. One of the main difficulties with local methods is selecting an appropriate support region Ω. If the support region is too small, it might not contain enough information to disambiguate non-matching targets, whereas overly large support regions tend to smooth out fine disparity variations. To address this problem, one approach would be to adaptively select an appropriate window size based on local variations in signal and disparity, which has been proposed by (Okutomi and Kanade, 1992; Kanade and Okutomi, 1994). These methods employ a statistical model of the disparity distribution within the window, enabling them to search for a window that produces

the disparity estimate with the least uncertainty. Another approach involves optimizing over a large class of arbitrarily shaped windows for each pixel-disparity pair, thereby applying pruning heuristics to make the search for an optimal window more efficient (Veksler, 2001).

2.4.3 Global algorithms

Global stereo matching algorithms usually involve formulating an optimization problem and the method that is used to find the global optimum. The computation of disparity gives rise to an ill-posed vision problem. This implies that in order for it to be solved, the optimization problem has to be constrained based on assumptions about the real world, such as the process of image formation or the smoothness of surfaces. The first part of this section deals with the formulation of the problem subject to its constraints, whereas the second part addresses the methods and algorithms that can solve these problems.

Formulating the optimization problem

Global methods are often formulated as an energy minimization problem. In that sense, energy is defined as the total matching cost of a disparity function and some smoothing term. The solution is found as the disparity function $d(x, y)$ that minimizes the global energy

$$E(d) = E_{match}(d) + \lambda E_{smooth}(d) \tag{2.15}$$

The matching energy variable quantifies how well the disparity function corresponds to the input image pair. The smoothness energy variable encodes the smoothness assumptions of the algorithm, contained in the disparity function. If the smoothness term is omitted, the maximum likelihood solution to the optimization problem as formulated consists of minimizing the squared error of the left (L) and right (R) image intensities

$$E_d = E_{match}(d) = \int\int \left[L(x, y) - R(x + d(x, y), y)\right]^2 dxdy \tag{2.16}$$

which corresponds to the traditional sum-of-squared-differences (SSD) metric (Anandan, 1989) often used to quantify matching cost in local methods. In order to obtain better results, many global methods add smoothness constraints. Poggio et al. (1985b) have observed that many early vision processes can be viewed as ill-posed problems that can be solved using regularization methods. They formulated the stereo problem in a standard Tikhonov regularization framework, with a variational principle that involved a convolution of the Laplacian of a Gaussian with the left and right image (E_{match}), and a Tikhonov stabilizer as

smoothness constraint (E_{smooth}):

$$E(d) = \int\int \left[\nabla^2 G * \left(L(x,y) - R(x + d(x,y), y)\right)\right]^2 + \lambda \left(\nabla d\right)^2 dxdy \tag{2.17}$$

The Tikhonov stabilizer penalizes large disparity gradients and makes d smooth everywhere, leading to poor results at object boundaries. It is interesting to note that for small disparity gradients, this global method approximates the local method detailed by Nishihara (1984) which used the sign of the Laplacian as the binary matching cost. A more general smoothing functional can be defined using two-dimensional spline surfaces

$$E_{smooth}(d) = \int\int \sum_{i=0}^{n} \binom{n}{i} \left[\frac{\partial^n d}{\partial x^i \partial y^{n-i}}\right]^2 dxdy \tag{2.18}$$

which have interesting physical interpretations, such as small deflection energy of a membrane ($m = 1$) or small deflection bending energy of a thin plate ($m = 2$) (Terzopoulos, 1986). The general problem with these types of quadratic smoothing functions is that they preserve continuity and thus, they are not suitable for reconstruction involving discontinuities (as they occur at object boundaries). One way of addressing this problem is by superposing splines of lower orders

$$E_{smooth}(d) = \sum_{n=0}^{m} \int\int w_n(x,y) \sum_{i=0}^{n} \binom{n}{i} \left[\frac{\partial^n d}{\partial x^i \partial y^{n-i}}\right]^2 dxdy \tag{2.19}$$

using weighting functions $w_n(x,y)$ that are allowed to be discontinuous, enabling the creation of discontinuities in the solution (Terzopoulos, 1986).

Probabilistic frameworks offer an alternative approach to the optimization problem that is encountered when applying global methods. Geman and Geman (1984) used Markov random fields (MRF) to model the smoothness of disparity. The prior probability of the disparity function can be modeled as a Gibbs distribution, with an energy E_p that encodes smoothness assumptions:

$$p(d) = \frac{1}{Z_p} \exp\left(\frac{-E_p(d)}{T_p}\right) \tag{2.20}$$

with Z_p being the partition function. Similarly, the measurement model can also be expressed as a Gibbs distribution with an energy function E_d which quantifies how well the disparity

function matches the input images x_L and x_R:

$$p(x_L, x_R|d) = \frac{1}{Z_d}\exp\left(-E_d(d, x_L, x_R)\right) \tag{2.21}$$

Using Bayes' rule, it follows that the posterior distribution is also an MRF:

$$p(d|x_L, x_R) = \frac{p(x_L, x_R|d)p(d)}{p(x_L, x_R)} = \frac{1}{Z}\exp\left(E(d)\right) \tag{2.22}$$

The maximum-a-posteriori (MAP) estimate will then correspond to the disparity function that minimizes the global energy (Szeliski, 1990):

$$E(d) = \frac{1}{T_p}E_p(d) + E_d(d, x_L, x_R) \tag{2.23}$$

with T_p being the temperature of the model of the disparity function. Now, it can be observed that the previously defined energy minimization framework for global methods corresponds to an example of optimal Bayesian estimation. Within this probabilistic framework, discontinuities can be modeled using line processes that impose constraints on the local spatial organization (Geman and Geman, 1984). Line processes can be understood as a set of discrete binary variables $\mathbf{l}$ defined for all nearest neighbor pairs $l_{i,j}$, which encode the presence or absence of discontinuities that can break the smoothing assumption. The new objective function is then optimized with respect to d and $\mathbf{l}$. Generally, line processes can be eliminated so that the problem simply reduces to a non-convex minimization over d (Blake and Zisserman, 1987). MRF models with line process have shown to reduce to standard regularization in certain cases. Thus, they do not necessarily need to be solved with stochastic algorithms (Black and Rangarajan, 1996).

A further improvement is achieved by making the terms in E_{smooth} depend on the spatial intensity gradient, meaning that the smoothing cost is reduced at the location of object boundaries (high intensity gradients). Gamble and Poggio (1987) developed a scheme involving coupled MRFs that integrate intensity edges with stereo depth to help discover discontinuities. Fua (1993) used a NCC-based local method to produce sparse disparity maps which were then interpolated with a global adaptive smoothing scheme, which preserved discontinuities at places where intensity edges were present. The observation that disparity discontinuities usually cause intensity edges appears to have a significant effect on the performance of some of the good global methods (Bobick and Intille, 1999; Boykov et al., 2001).

Solving the optimization problem

A variety of algorithms exist which aim to solve the formulated global optimization problems. Variational calculus can be applied to produce a partial differential equation of the unknown disparity function $d(x, y)$, which can be solved numerically in an iterative scheme. The problems usually associated with this approach are slow convergence and getting stuck on local minima (in the case where the functional is non-convex). The graduated non-convexity (GNC) algorithm addresses this problem by first solving a convex approximation, and then iteratively moving towards the non-convex functional, using the result of one optimization as the starting point for the next (Blake and Zisserman, 1987). Traditional stochastic algorithms that are associated with MRFs are typical Markov chain Monte Carlo (MCMC) methods such as the Metropolis algorithm (Metropolis et al., 1953) or the more efficient Gibbs sampling algorithm (Geman and Geman, 1984). Local minima are often avoided using the concept of simulated annealing (Geman and Geman, 1984; Barnard, 1989). A deterministic approximation of stochastic simulated annealing based on mean-field theory is the mean-field annealing technique, described in Geiger and Girosi (1989), which averages the statistics of the annealing process. Yet another more efficient method is based on graph theory (Roy and Cox, 1998; Boykov et al., 2001; Kolmogorov and Zabih, 2001). Graph-cut algorithms formulate the energy minimization problem as a question of finding the maximum flow in a graph. In the case of stereo correspondence, each point in disparity space (x, y, d) represents a node of the graph connected to its nearest neighbors by weights which depend on the matching cost of a potential target at that location (edges in direction of d) and smoothness assumptions (edges in direction of x and y). The nodes at disparity $d = 0$ are connected to the source node s. Likewise, the nodes at the maximum disparity are connected to the sink t. Determining the maximum flow from s to t is equivalent to finding the minimum cut that separates the graph, such that no flow can pass from s to t (*max-flow-min-cut* theorem). This cut represents the disparity surface, which is the solution to the energy minimization problem. Graph-cut methods are efficient approximation algorithms when tackling NP-hard, discontinuity-preserving energy minimization problems (Veksler, 1999). More recently, stereo matching based on belief propagation has been introduced (Freeman et al., 2000; Sun et al., 2003). Belief propagation is an efficient message-passing algorithm which calculates the marginal distribution of hidden nodes in Bayesian networks such as MRFs. In Sun et al. (2003) a Markov network, consisting of three coupled MRFs that model smoothness, depth discontinuity and occlusion is used in conjunction with the belief propagation algorithm to find the MAP estimation in the Markov network. Finally, Szeliski et al. (2008) compare the most prominent energy minimization methods for MRFs with smoothness-based priors.

Another class of global methods is based on the concept of dynamic programming. It was first used to search for correspondences between edges on single epipolar lines (referred to as *scanlines* in the original literature), whereby the problem was conceived as a path-finding task on a 2D plane (Baker, 1982). The concept of dynamic programming is based on breaking down a complex problem into a set of smaller sub-problems. Once a sub-problem is solved, the solution is memorized. Dynamic programming can be applied when the overarching

problem consists of repeated sub-problems and the optimal solution can be composed from the optimal solutions of the sub-problems. A typical example where dynamic programming is very efficient is finding the shortest path between two points. Dynamic programming can find the global minimum in polynomial time, avoiding exponential combinatorial complexity. The path-finding problem can be directly applied to two-dimensional stereo vision, whereby the path sought corresponds to the one-dimensional disparity line and the cost of potential matches (that are visited on the path) is minimized rather than their length. Ohta and Kanade (1985) have successfully applied dynamic programming to a three-dimensional environment by using two, coupled dynamic programming processes that run simultaneously. The first process solves the matching problem (intra-scanline search) while the second optimizes the vertical consistency of disparity lines (inter-scanline search). The disadvantage of dynamic programming is that it imposes an ordering constraint (Yuille and Poggio, 1984), meaning that edges on the left and right must have the same order (which might not be the case in certain scenes that have narrow objects in the foreground, for example). Belhumeur (1996) have proposed a Bayesian approach to stereo which uses dynamic programming to obtain and model characteristics related to image formation such as depth, surface orientation, object boundaries and surface crease. Other dynamic programming approaches which have been developed include those which focus on dense disparity maps, search for matches and occlusions simultaneously (Cox et al., 1996; Bobick and Intille, 1999), and deal with the problem of inter-scanline consistency (Birchfield and Tomasi, 1999).

2.4.4 Iterative algorithms

One class of iterative methods to compute stereo disparity are the so-called hierarchical algorithms, that typically work on image pyramids. Firstly, rough disparity values are computed from coarser levels of the image hierarchy. Subsequently, they are used to restrict the search at finer levels in an iterative manner. Quam (1984) have demonstrated that by using coarse disparity values which have been estimated in advance, the input images can be warped to improve the performance of the extensive matching process afterwards. Witkin et al. (1987) applied a coarse-to-fine matching technique in a constrained variational optimization framework. In general, hierarchical approaches have been used by various researches due to their computational efficiency (Bergen et al., 1992). Another class of iterative algorithms are those inspired by the computational models of human stereo vision (Marr and Poggio, 1976; Drumheller and Poggio, 1986; Mahowald and Delbrück, 1989; Zitnick and Kanade, 2000) that have already been discussed in Section 2.3.

2.4.5 Evaluation of stereo algorithms

Scharstein and Szeliski (Scharstein and Szeliski, 2002) developed an evaluation framework using test images and accurate ground-truth disparity maps that became widely accepted as the standard performance assessment technique for dense two-frame stereo vision algorithms. Two main quality metrics are used to compare the computed disparity map $d(x, y)$, with

a ground truth disparity map $d_T(x,y)$. The first, the root-mean-squared (RMS) error, is computed as follows:

$$RMS = \frac{1}{N} \sum_{(x,y)} \sqrt{\left(d(x,y) - d_T(x,y)\right)^2} \tag{2.24}$$

where N is the total number of pixels. The second, the percentage of bad matching pixels (BMP), can be expressed as

$$BMP = \frac{1}{N} \sum_{(x,y)} \left(\left|d(x,y) - d_T(x,y)\right| > \delta_d\right) \tag{2.25}$$

where δ_d is the disparity error tolerance. The most common evaluation metric used in the literature is the BMP with a disparity error tolerance of $\delta_d = 1.0$.

2.4.6 Recent development

Most of the best contemporary approaches to stereo vision are based on the fundamental principles discussed in the previous sections. During the last decade, a variety of sophisticated algorithms have been proposed, which have optimized and evolved these techniques. Probably the most important characteristic of the newer methods that achieve state-of-the-art performance is their ability to properly handle discontinuities and occluded areas (Egnal and Wildes, 2002). By introducing adaptive support weights (Yoon and Kweon, 2006), the accuracy of local methods has been significantly improved, making them comparable to global methods which traditionally performed far better. Another important feature of more recent algorithms is that color information can be used. Some local methods based on an adapted SAD metric which considers color information are described in Mühlmann et al. (2002) and Yoon and Kweon (2006). In addition, global methods often include color information within the concept of image segmentation. Image segmentation enables the global minimization problem to be formulated on a segmental, rather than a pixelar level, which leads to more robust results in untextured areas and allows precise localization of depth boundaries (Bleyer and Gelautz, 2005; Hong and Chen, 2004; Zitnick and Kang, 2007). Other popular global approaches either directly incorporate color segmentation within the minimization framework as a further constraint (Sun et al., 2005), or simply adapt the energy cost function to consider color similarity (Yang et al., 2009). The *semi-global matching* (SGM) technique improves upon the basic dynamic programming and scanline optimization approaches (Hirschmuller, 2005, 2008). Instead of solving the matching problem for one-dimensional horizontal scanlines, the SGM method uses multiple one-dimensional scanlines tilted in all directions and computes the final disparity map by selecting the minimum of the summed cost vectors from the individual results. The SGM method provides a very efficient approximation compared to the usual 2D optimization problem and can be solved in polynomial time. The most recent algorithms that set

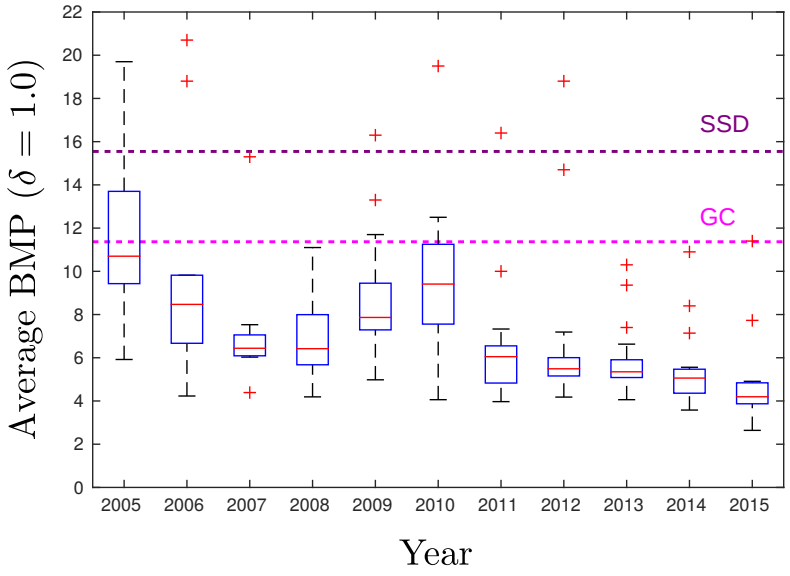


Figure 2.7: Development of stereo algorithm performance over the last decade. The benchmarks of the traditional SSD and basic graph-cut (GC) method are shown for reference. Both methods were implemented in their simplest form and evaluated by Scharstein and Szeliski (2002). In total, 165 stereo algorithms, listed in the Middlebury stereo evalutation list, were considered.

the standard for state-of-the-art performance often focus on improving the matching costs to make them robust against image noise and textureless regions. Others have adopted disparity refinement techniques, which have the drawback of being computationally expensive. While these techniques are not relevant enough to this thesis to justify more detailed explanation, it is interesting nevertheless to note how the performance of the algorithms has improved since the introduction of the basic methods. Fig. 2.7 shows the evolution of performance for stereo algorithms, whereby the sorting metric used corresponds to the average BMP achieved on the Middlebury stereo dataset, which can be found online[2]. For comparison, the performance of the traditional SSD and basic graph-cut (GC) methods are also indicated. In general, a steady increase in performance can be observed to date, and the best state-of-the-art methods now reach a BMP below 3%. Interestingly, the average performance started to decrease again in 2008. This is because at that time, research efforts became increasingly focused on real-time performance rather than advancing the limits of the matching performance.

2.4.7 Real-time stereo vision

An important issue that has not been addressed so far is the time these stereo algorithms need to compute depth. During the last decade, the perception of depth became an increasingly important aspect of autonomous robotics and mobile devices. Perceiving in depth is a basic requirement for solving some of today's biggest challenges such as robot navigation or virtual and augmented reality. Detailed reviews that investigate the suitability of stereo vision algo-

[2] http://vision.middlebury.edu/stereo/data/

rithms for resource-limited systems can be found in Lazaros et al. (2008) and Tippetts et al. (2013). This section provides a brief summary of the real-time performance of state-of-the-art stereo vision algorithms with respect to the associated hardware systems. The most common metric for evaluating runtime performance is the number of millions of disparity evaluations per second (Tippetts et al., 2013):

$$\mathrm{Mde/s} = \frac{W \times H \times D}{t} \cdot \frac{1}{1'000'000} \tag{2.26}$$

where W is the width, H the height and D the number of disparity levels. When a stereo vision system is termed "real-time", this is not necessarily a precise quantitative statement. It merely means that the system processes the input data in real-time, which of course strongly depends on the rate at which data is inputted. In fact, here it is assumed that real-time performance corresponds to a frame rate of 30 Hz with a QVGA (320×240) resolution and 32 disparity levels which would be roughly equivalent to 70 Mde/s. Among the fastest real-time stereo vision algorithms are the local and semi-global methods that apply very fast bilateral cost-volume filtering. Such algorithms can reach up to a few hundred Mde/s on a modern GPU, while still achieving near state-of-the-art matching performance (Yu et al., 2010; Rhemann et al., 2011; Zhang et al., 2011; Wang et al., 2012; Kowalczuk et al., 2013). The same algorithms are about two order of magnitudes slower on a CPU. Some of the best GPU-accelerated stereo matching performance in terms of accuracy and speed which is currently possible is achieved using the hardware-efficient bilateral filtering method described in Yang (2014). This approach achieves 67 Fps with a BMP of 5.4%. Some GPU-accelerated methods report up to several thousand Mde/s but these numbers are only achieved on high-resolution images which enable multi-scale schemes to be exploited to improve the computational efficiency. Drazic and Sabater (2012) present a multi-scale algorithm based on a modified version of the SSD. They achieve near real-time performance with full-HD (1920×1080) images at a frame rate of 12 Hz. In contrast, global methods generally yield more accurate results, but are significantly slower. A real-time implementation of the hierarchical belief propagation algorithm achieved 20 Mde/s, corresponding to 16 frames per second with QVGA resolution and 16 disparity levels (Yang et al., 2006). The semi-global method presented in Wang et al. (2012) applies adaptive cost-volume filtering in a dynamic programming framework to achieve a remarkable 90 Mde/s. A similar method combines local cost aggregation with a low-complexity iterative refinement technique and reports comparable runtime performance at an average BMP of 6% on the Middlebury stereo dataset.

By deploying stereo algorithms on dedicated hardware, runtime performance can be significantly accelerated, which has led to a growing interest and shift towards such systems. The drawbacks are that they tend to be technically difficult to implement, comparatively inflexibile, and highly expensive in terms of time and manufacturing cost (in the case of ASICs). A good trade-off can be reached by using FPGAs. One of the first FPGA stereo vision systems, described in Jia et al. (2003), used a SAD method with a fixed window size which

achieved more than 30 Fps with VGA (640 × 480) resolution back in 2003. An improved version that employs adaptive window sizes and a cellular automata method to remove noise from the disparity map is presented in Georgoulas et al. (2008). The proposed method computes the disparity map very rapidly (1090 Fps for QVGA and 275 Fps for VGA) at the expense of accuracy (BMP≈10% at a typical pixel coverage of 60%). Further local methods that have been implemented on FPGAs combine SAD with gradient-based census transformation (Ambrosch and Kubinger, 2010), and check consistency to eliminate unreliable disparity values (Zicari et al., 2012). The latter processes 1280 × 720 grayscale images to produce disparity maps with 30 disparity levels at a rate of 97 Hz, which corresponds to about 2,700 Mde/s. Global methods for FPGAs have been mainly implemented using dynamic programming approaches (Park and Jeong, 2007; Kalomiros and Lygouras, 2010). Unsurprisingly, as with stereo processing on dedicated hardware, a comparative study shows that dynamic programming yields more accurate results than local SAD, at the cost of reduced speed (Kalomiros and Lygouras, 2010).

Only a few studies have addressed stereo processing on ASICs, due to the obvious drawbacks such as long prototyping times and high production costs. These designs were mainly restricted to using simple local methods (Kuhn et al., 2003; Hariyama et al., 2004; Philipp, 2009) but exhibited superior performance, power consumption and effective size nevertheless. For example, Kuhn et al. (2003) have presented a design that achieved 54 Mde/s, integrated in a 0.25 μm standard CMOS technology occupying an area of less than 3 mm^2.

2.5 Discussion

Binocular vision was among the very first problems addressed in physiology. The discovery of cells that were tuned to disparity in the striate cortex has led to an extensive investigation that began more than 50 years ago and has continued till the present day. One of the main breakthroughs in this research was the observation that disparity-tuned cells can be described as binocular energy neurons. While this model describes a vast amount of cells in V1 very well, it does not explain how ambiguity among the stimuli is resolved and thus, how the stereo correspondence problem is solved. It is often argued that the correspondence problem is addressed in higher visual areas, which is supported by evidence of V1 neurons that signal disparity which is not perceived as depth. However, more and more cells in V1 have been found which exhibit behavior that deviates from the predictions of the classical disparity energy model. Among such contradictions are the attenuated response to anti-correlated RDS, for example, or the selectivity to natural stimuli (it can be shown that energy neurons respond optimally to stimuli that do not occur in the real world). This began a still unresolved debate about where the correspondence problem is solved in the brain and the role of V1 neurons in that process. A generalization of the energy model accounts for many of the deviations from the classical energy model observed in recordings of real cells. The generalized disparity energy model suggests that V1 neurons play a more significant role in the resolution of the correspondence problem than previously assumed. Although disparity detectors are very well described on a functional level, it is not yet clear how the model is implemented, and by

which neural mechanisms. Furthermore, it is not clear if the rate-based energy model can explain all the aspects of real spiking disparity neurons. In particular, the temporal dynamics of the stimuli and disparity neurons might account for important mechanisms of stereopsis that are not described by the energy model. Despite the vast amount of evidence supporting the interaction of motion and disparity, temporal aspects have not been studied at length. This shortcoming is particularly highlighted by the lack of a comprehensive understanding of certain psychophysical illusions, such as the Pulfrich phenomena.

Computing depth from a pair of images has been a long standing challenge in machine vision which is still not regarded as being fully resolved. An enormous variety of methods exist which can reliably solve the correspondence problem, even for very complex scenes. One of the main challenges in this process is handling occlusions and disparity discontinuities properly. Some state-of-the-art global methods can produce very precise disparity maps with an average percentage of correct matches of 98% when tested on a standardized stereo data set comprising complex scenes with occlusions. Typically, such benchmarks have only been achieved at a very high computational cost, which made them unsuitable for real-time processing. However, recent local and semi-global methods are only slightly less accurate, but significantly more efficient. In the last decade, research has been increasingly focused on developing real-time systems, as depth perception and 3D imaging have become increasingly important for robotics and mobile devices. When deployed on dedicated hardware such as GPUs, FPGAs or even ASICs, stereo vision systems can achieve impressive frame rates, often far beyond the classical video rate of 30 Hz. However, to a large extent, these systems involve power-hungry GPUs or FPGAs, which makes them difficult to integrate in mobile platforms. Similarly, as the main initial goal was to improve accuracy regardless of real-time performance, there has been barely any focus on the power consumption of these systems. Addressing this issue is essential to the future development of robots, autonomous systems and mobile devices. Mainly for this reason, stereo vision remains an active field which promotes innovative interdisciplinary research in areas such as neuromorphic engineering, which promises great potential to address the unresolved issues.

Stereopsis has made a definite impact on machine vision. For example, the early cooperative algorithms are nowadays used as relaxation processes to solve complex optimization problems in stereo vision algorithms. Conversely, methods from machine vision such as matching based on normalized-cross-correlation have been used to explain certain aspects and limits of human stereopsis. Another example of the strong interaction between these two fields is the disparity gradient limit, which describes a constraint observed in human perception, and is similar to the ordering criterion applied in stereo vision algorithms. Recently, the fields have evolved increasingly independently, and little effort has been made to coordinate the disciplines. Following on from the intention of the early cooperative stereo algorithms, this thesis attempts to reunite the disciplines of neuroscience and machine vision. Obviously, this endeavor starts with cooperative processes.

3 From Neuromorphic Hardware to Event-based Machine Vision

3.1 Understanding brain-inspired computation

The effortless behavior of animate beings since the beginning of time in complex and continuously changing environments has strongly influenced human technology. In this context, a major human goal has been to build intelligent systems in the form of computers. Computer technology has progressed exponentially in the twentieth century to the extent that today, computers are used everywhere. Despite the remarkable capabilities of computers, however, animal brains continue to significantly outperform them by multiple orders of magnitude when it comes to efficiency. To illustrate this with an example, a honey bee can simultaneously perform a variety of complex behaviors involving acrobatic flight maneuvers, visual landmark recognition, navigation, communication and foraging. All this is performed on a tiny biological substrate, which is about $1\,\mathrm{mm}^3$ in volume and requires less than $10\,\mu\mathrm{W}$ of power. While these tasks may appear trivial to humans, they represent a huge challenge for artificial systems, such as self-driving cars for example. If the energy cost of synaptic activation is compared with a *multiply-accumulate operation* (MAC) in digital silicon, it can be observed that brains are more power efficient than computers by a factor of about one million. This huge difference does not derive from some more efficient physical processes that are employed by nature. Instead, it stems from the fundamentally different principles of brain technology. Computation in the brain is distributed over billions of interconnected neurons that operate in parallel. Each neuron connects to thousands of other neurons through modifiable synapses. Neurons largely communicate using stereotypical action potentials, which means that the communication is digital. In contrast, action potentials are processed in the soma of neurons in an analog manner. Neurons are self-timed, analog processors that have low resolution and exhibit adaptive behavior. Self-timing is an important principle of brains which implies that processing only happens when data is available. Another important concept related to brains is that the memory stored in synapses is distributed and co-localized with the processing. These are all mechanisms that are fundamentally different from those of computers. A computer is based on the principle of separating the processing unit from the memory, which then continuously communicate with each other. The processing unit

processes data sequentially at very high rates. The entire process of computation is based on logic state machines which are digital, bit-perfect and deterministic. The whole system is synchronized with a fast global clock and once such devices are produced, their capabilities are fixed from that point forward. To compete with the huge bandwidth of brains (achieved through massively parallel processing), computers have to employ clock rates that are orders of magnitude higher than the firing rates of neurons. Indeed, this disparity is the main cause of the inferiority of computers in terms of power efficiency.

Beyond power efficiency, however, another distinguishing characteristic of brain technology is its ability to adapt and learn. Rather than being programmed, the neural networks in the brain are *trained* by real-world examples. Over the course of evolution, a set of rules have emerged that have produced neural networks that can develop intelligent human behavior when exposed to real-world stimuli over years. Neural computation is *self-contained*, in that the meaning of data is intrinsic to the substrate that processes it. This is an important prerequisite for the emergence of intelligent behavior because it means that the computation the network entails is determined by the network itself rather than a human programmer who defines symbolic encodings of the data, as is the case for computer algorithms. Unlike neural networks, the behavior of the computer is deterministically described by a set of rules that define how these abstract symbols are processed. The computer has no understanding of the meaning and significance of the symbols. This fundamental insight has finally led to machines becoming more intelligent and in the past decade, major breakthroughs in artificial intelligence were achieved. Among these are self-driving cars (Thrun et al., 2007) or computers that can beat human players in chess (Hsu, 2002) or 'Jeopardy!' (Ferrucci et al., 2010).

3.2 Neuromorphic engineering in a nutshell

In the late 1960s, researchers observed that the ion-channels found in nervous systems behave similarly to electronic transistors. In particular, it was shown that in the case of both ion-channels and transistors (in the *weak-inversion* region), conductance is exponentially dependent on voltage. This piqued the interest of Carver Mead, a professor at Caltech, who started to study the analogy between the biophysics of nervous systems and the physics of silicon. As a result of this research, Mead propagated the idea of building entire systems in silicon based on the organizing principle used by the nervous system. This idea served as the foundation for the field of *neuromorphic engineering*. Many of the fundamental concepts and circuits are described in Mead's seminal book on analog very-large-scale-integration (aVLIS) (Mead, 1989), which were further developed by his students. This led to the development of a widely recognized discipline encompassing a growing academic community spread throughout the entire world. In the following section, a brief introduction of the key concepts of neuromorphic engineering is provided. A detailed summary of neuromorphic design principles, circuits, applications and future challenges can be found in books by Liu et al. (2002) and Liu et al. (2015).

3.2.1 Address-event representation

One of the main challenges of neuromorphic systems is communication. Nature has built the brain in 3D, providing enough space so that a single neuron can connect to thousands of others. Conversely, integrated circuits have thus far been fabricated on 2D silicon surfaces, whereby the typical units (logic gates) are connected to three or four other units. Fortunately, the bulk mobility of electrons is about 10 million times higher than that of ions. This enables communication among many units in silicon by applying the concept of *time-multiplexed* shared buses. The nervous system communicates with stereotypical spikes which can be modeled as digital events. Often, in models of neural computation, the amount of information a spike carries is limited to the time of the spike, i.e. the time it occurred relative to other spikes in the system. Neuromorphic engineering has come up with an elegant way of representing time, which is commonly described as the idea of "time representing itself". This simply means that spikes are communicated in the form of digital events which are sequenced in the correct order and at the precise time they should be delivered. This concept can only be implemented if the communication fabric is not overloaded, in order to avoid delays and enable the system to run in real-time. Considering the low spiking frequency of neurons (tens of Hz) compared to the switching frequency of digital electronics (GHz), however, this seems feasible even for a high degree of multiplexing. Furthermore, it can be observed that in biological systems, small variation in the timing of neuron spikes is not critical.

In response to these considerations, the *address-event representation* (AER) standard for communication in neuromorphic systems has been proposed (Lazzaro et al., 1993; Mahowald, 1994a; Sivilotti, 1990). The AER protocol is based on a packet-switched communication scheme, whereby digital events (packets) are send over a time-multiplexed shared bus. Typically, an AER link exists between two populations of cells as shown in Fig. 3.1a. Whenever a sender cell (usually a neuron) spikes, it produces a digital event. Each event encompasses the address of its source (encoded in a string of bits) and the time of occurrence. Rather than being explicitly encoded in the event, time represents itself as explained above. The communication comprises two sites. At the site of the sender, multiple neurons are multiplexed onto a single communication channel whereas at the site of the receiver, AER events are demultiplexed into individual spikes that address different synapses. Multiplexing requires an encoding circuit (AER encoder) and demultiplexing requires a decoding circuit (AER decoder). Since multiple spikes can occur at the same time at the site of the sender, an additional arbitration circuit is required that selects the order of access to the shared communication channel. A traditional encoder uses $\log(N)$ addressing bits for N neurons. If a non-trivial or flexible mapping between the sender and receiver is required, an additional mapping circuit can translate source addresses to target addresses. Note that in the events, the source address is encoded at the site of the sender, whereas the target address is encoded at the site of the receiver.

Typically, AER circuits are implemented using *asynchronous* logic. To avoid incomplete data transmission, a *four-phase bundled data* handshake protocol is often used. The handshaking procedure is illustrated in Fig. 3.1b. If a neuron on the sender site fires, it writes its address

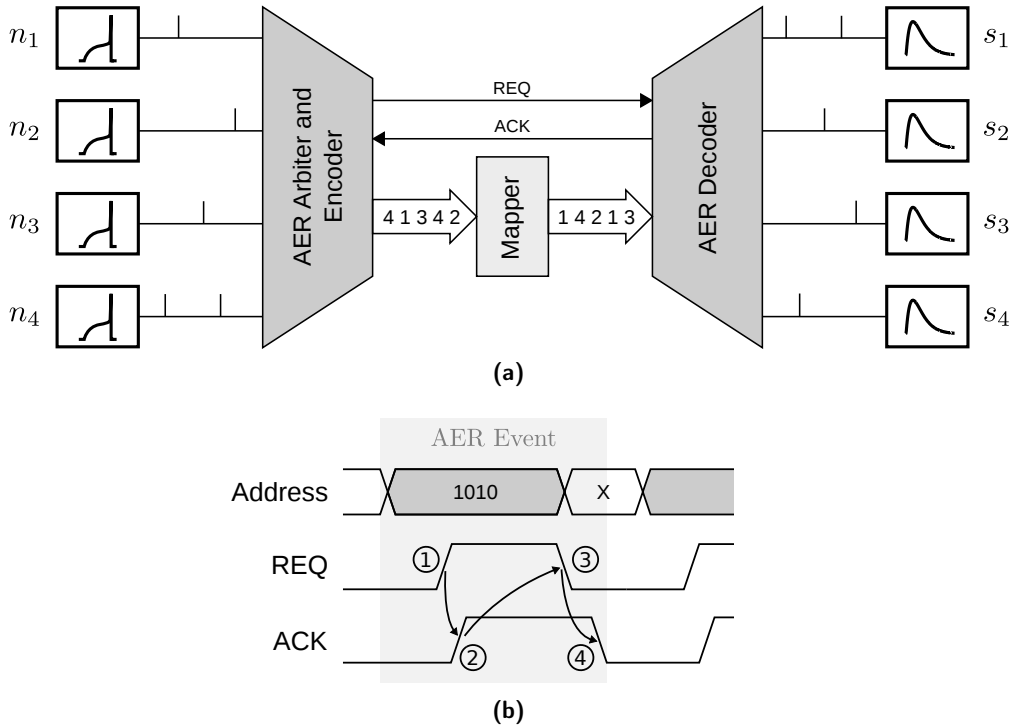


Figure 3.1: Address-event representation (AER) communication. **(a)** Spikes from a pool of source neurons ($n_1 - n_4$) are multiplexed by an arbiter and encoder onto a single communication channel. On the receiver site they are demultiplexed by a decoder circuit and delivered to the target synapses ($s_1 - s_4$). **(b)** AER handshaking with an asynchronous, four-phase, bundled data protocol. Request (REQ) and acknowledge (ACK) are active-high signals.

onto the data bus as soon as it is selected by the arbiter. Once the data is validated, the arbiter sends a request by raising the REQ line. The cell that receives the event (usually a synapse) responds with an acknowledgement signal. Once the acknowledgement has been received by the sender, it releases the data and re-acknowledges by removing the request. At this point, the receiver releases the ACK line and the transmission is completed.

3.2.2 Self-timed and analog processing

Neuromorphic engineering adopts two important principles from brain technology: *self-timing* and *analog processing*. The majority of today's integrated circuits are *synchronous*, which means that their state changes in sync with a global clock signal. This has the advantage that after a pulse of the clock, the circuit has time to propagate to a stable state before the next pulse occurs. This makes synchronous circuits easy to design and their behavior can be simply described with a hardware description language (HDL) that can be subsequently automatically translated into a circuit design. The convenience of the design flow of synchronous circuits comes at the expense of wasteful power consumption, however. States change continuously because of the uniform clock signal, even if no data needs to be processed. Unlike synchronous circuits, the brain does not employ a global clock signal but is *self-timed*, mean-

ing that neurons process spikes at the times they are delivered. Integrating the concept of self-timing into electronic circuits, using so-called *asynchronous* circuits, has proven to be a very difficult challenge. This is mainly due to the sensitivity of such circuits to data arriving at different times, which can lead to incorrect states being triggered. Asynchronous circuits are an active research area in which the major breakthroughs have been the first competitive asynchronous microprocessor (Martin et al., 1997), the first automated synthesis for asynchronous FPGAs (Peng et al., 2005), and the first automated layout for asynchronous circuits (Karmazin et al., 2013). In neuromorphic systems, asynchronous logic synthesis is typically used to design AER circuits (Boahen, 2000). Recently, a million spiking-neuron integrated circuit was completely designed using digital asynchronous logic and it currently holds the record for the lowest operational power consumption in neuromorphic electronics (Merolla et al., 2014).

Another important aspect of neuromorphic engineering is *analog computation*. The idea of analog computation is to make use of device physics to model the problem being solved. This can be very efficient at carrying out low-precision processing because the computational primitives arise from the physics of single computing devices (such as the integrating behavior of a capacitor) or a value is conveyed with a single wire (such as an analog current). Such computation is prone to error, however, due to mismatch of device parameters and thermal noise. In addition, unlike digital computing systems, the signals are not restored at each stage of the computation. These caveats notwithstanding, analog computation can be more power efficient and requires less resources. In a review of the pros and cons of analog and digital computation, Sarpeshkar (1998) concludes that the most efficient form is an intimate mixture of both. He also suggests that one of the reasons for the efficiency of the human brain, is its hybrid architecture. Neuromorphic systems often use analog circuits to model the dynamics of neurons and synapses, or the front ends of sensors. Although there are ways to compensate for device mismatch, either on an algorithmic level (Neftci and Indiveri, 2010) or at the circuit level (Hasler and Lande, 2001; Hasler and Marr, 2013), mismatch is not always an undesired effect and can also be exploited (Sheik et al., 2012; Yao et al., 2013).

3.3 The silicon retina

The human retina is not simply a sheet of photo-sensitive cells that sends images to the brain. It is actually a rather complex neural network that performs a considerable amount of visual processing. When the vast amount of photoreceptors in the human retina (about 120 million rod cells and 6 million cone cells) are compared to the capacity of the optic nerve (the communication channel between the retina and the brain), it seems evident that the retinal network *decorrelates* the perceived visual input and only transfers useful information to the brain. This compression is achieved by retinal ganglion cells that encode spatial and temporal contrast. Another important mechanism in the retina is *adaptation*, which is responsible for the striking ability of the visual system to reliably function under various lighting conditions. It has been observed that the retina responds stronger to changes in illumination than to steady illumination. The responses were found to be constant to the illumination on a log

scale, which effectively means that the retina responds to contrasts in illumination regardless of the degree of background illumination (Normann and Perlman, 1979). The mechanisms of light adaptation have been studied extensively (Shapley and Enroth-Cugell, 1984; Rieke and Rudd, 2009), but there are many other forms of adaptation in the retina. The retinal network includes over 50 distinct cell types which perform a variety of complex computational tasks ranging from contrast adaptation to various forms of motion detection (Gollisch and Meister, 2010).

Neuromorphic vision chips that replicate some of the principles found in the human retina are called *silicon retinas*. The aim of such chips is to provide more efficient and better performing visual sensors by applying the concepts of adaptation and data compression. Silicon retinas can be classified based on the way they encode visual information. *Spatial contrast* sensors encode differences in spatial intensity, whereas *temporal contrast* sensors encode changes in temporal illumination. Although both types of sensor reduce the level of redundancy, the former is better suited to analyzing *static* scenes, in order to recognize objects, for example. The latter is more applicable to *dynamic* scenes, in which objects need to be tracked. The first silicon retina that was developed in the context of neuromorphic engineering was built by Mahowald and Douglas (1991). It has taken more than two decades since this early prototype to finally produce practical devices that can be used for real-world applications. Today's silicon retinas are also called *event-based cameras* because of their spiking AER output. In the following subsections, two sensors that were extensively used in this thesis are explained in greater detail.

3.3.1 Dynamic vision sensor

The *dynamic vision sensor* (DVS) is an asynchronous temporal contrast sensor that communicates changes in temporal intensity in the form of an event stream which is output through an AER interface (Lichtsteiner et al., 2008). The informative spiking output reduces redundancy while also preserving precise timing information. Each pixel independently responds to relative changes in intensity. Thus, effectively, they encode changes in the reflectance of the scene regardless of background illumination. This is achieved through a logarithmic conversion carried out by the photoreceptor followed by a differencing operation. As a result, the response encodes *temporal contrast* which is defined as (Lichtsteiner et al., 2008):

$$TCON = \frac{1}{I(t)} \frac{dI(t)}{dt} = \frac{d}{dt} \ln(I(t)) \tag{3.1}$$

where $I(t)$ is the time-dependent photo current. The sensor comprises a 128 × 128 pixel array and has a wide dynamic range of over 120 dB and a very low power consumption of 23 mW. The minimum latency of 15 μs is achieved at a pixel illumination of 1 klux. The DVS facilitates an excellent balance between power consumption, latency and dynamic range and it has been successfully applied to a wide range of applications including various tracking tasks (Delbruck

and Lichtsteiner, 2007; Drazen et al., 2011; Ni et al., 2012), gesture recognition (Lee et al., 2012) and visual positioning in drones (Mueggler et al., 2014), to name just a few.

Pixel design

The simplified circuit schematic of the DVS pixel is shown in Fig. 3.2a. It comprises a logarithmic photoreceptor, a differencing circuit and comparators. The front end of the photoreceptor makes use of a transimpedance configuration that logarithmically converts the photo current into an amplified voltage. In turn, this voltage is used as feedback to clamp the photo diode at a virtual ground (Delbruck and Mead, 1994). In comparison to a passive configuration, this active feedback control enables a higher bandwidth by a factor proportional to the gain of the amplifier. To improve the power consumption, this circuit also applies adaptive biasing of the amplifier (Delbruck and Oberhoff, 2004). This self-biasing sets the speed of the amplifier at an optimal rate according to the illumination level. The illumination level is measured as the sum of all photo currents ΣI. The output of the photoreceptor circuit is a continuous-time logarithmic voltage of the photo current

$$V_P(t) \propto \ln\left(\frac{I(t)}{I_0}\right) + K \tag{3.2}$$

where I_0 is a constant and K is an offset. Due to variation in the transistor threshold, this output is substantially mismatched between pixels. The mismatch is removed by the capacitive differencing circuit that resets the output after the generation of an event. The differencing circuit contains an inverting amplification and its output is directly related to the temporal contrast $TCON$

$$\begin{aligned} \Delta V_D &\propto -A \cdot \Delta V_P \\ &= -A \cdot \ln\left(\frac{I(t+\Delta t)}{I(t)}\right) \\ &= -A \cdot \Delta \ln\left(I(t)\right) \\ &= -A \int_t^{t+\Delta t} TCON(\tau) d\tau \end{aligned} \tag{3.3}$$

where $A = C_1/C_2$ is the gain of the capacitive amplifier. Although the differencing circuit integrates TCON, however, since TCON is the derivative of log intensity, the result is a direct amplification of the change in log intensity since the last reset. Since the amplification is determined by the well-matched capacitor ratio C_1/C_2, the mismatch is significantly reduced compared to the predecessors of this chip (Kramer, 2002; Lichtsteiner et al., 2004). The final stage compares ΔV_D to a contrast threshold θ, which involves separate units signifying an increase (ON) and decrease (OFF) in contrast. At the point when the threshold is reached,

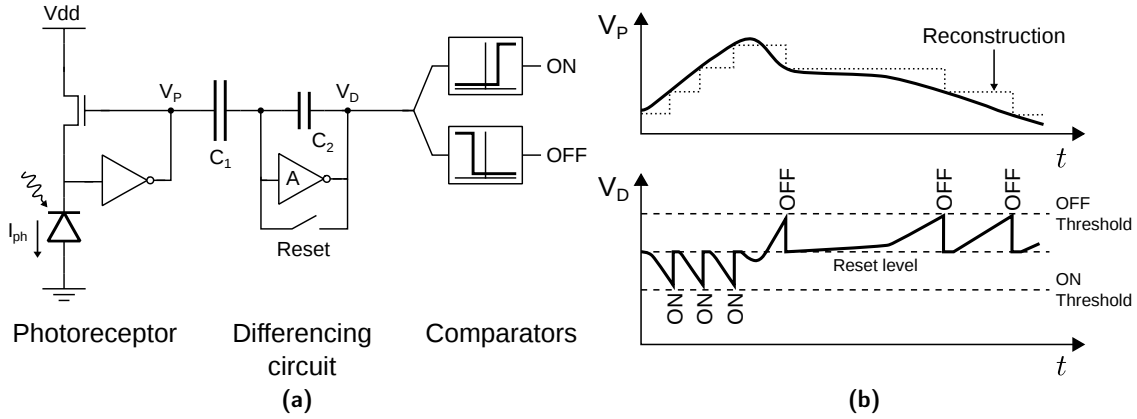


Figure 3.2: The dynamic vision sensor (DVS) pixel. **(a)** Simplified pixel design comprising photoreceptor, differencing and comparator circuits. **(b)** Principle of operation. Adapted from Lichtsteiner et al. (2008).

either an ON or OFF event is generated. The principle of operation is illustrated in Fig. 3.2b. Each event corresponds to a change in log intensity of size θ. The event rate that is generated can be approximated by

$$R(t) \approx \frac{TCON}{\theta} = \frac{1}{\theta}\frac{d}{dt}\ln(I(t)) \tag{3.4}$$

Modified versions of the DVS pixel enhance contrast sensitivity by including an extra preamplfication stage. However, this comes at the cost of increased power consumption and reduced dynamic range (Lenero-Bardallo et al., 2011; Serrano-Gotarredona and Linares-Barranco, 2013).

3.3.2 Asynchronous time-based image sensor

The *asynchronous time-based image sensor* (ATIS) (Posch et al., 2011) combines event-based change detection with an absolute intensity measurement. Each of the 304×240 pixels individually initiates an intensity measurement after a local change in brightness has been detected. Ideally, this leads to a situation where temporally redundant information is entirely suppressed and the video can be compressed losslessly at a pixelar level. The change detector circuit is congruent with the DVS pixel. Upon completion of an intensity measurement, the grayscale value is asynchronously communicated off-chip through an AER output. One of the main innovations of the ATIS is the way that intensity is temporally encoded based on a pulse-width-modulated (PWM) intensity measurement. This innovative concept is called time-domain correlated double sampling (TCDS) (Matolin et al., 2009). The principle is illustrated in Fig. 3.3b and explained in the remainder of the present subsection. The simplified schematic of the pixel design is shown in Fig. 3.3a. Once a change in intensity has been detected by the DVS pixel, the intensity measurement is initiated. During this measurement, a capacitor is

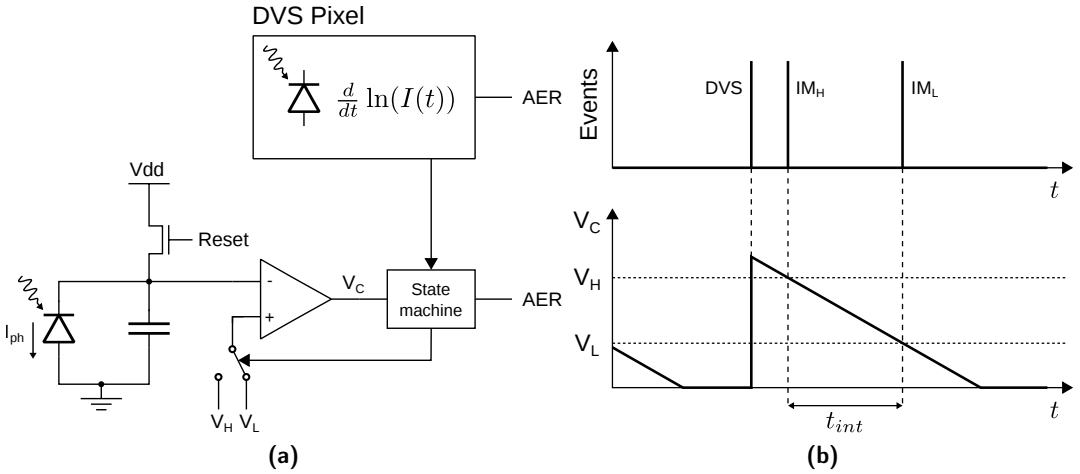


Figure 3.3: The asynchronous time-based image sensor (ATIS) pixel. **(a)** Simplified schematic of the pixel design. The in-pixel state machine controls the switch for the two voltage comparisons. **(b)** Principle of operation. After a DVS event, two intensity measurement events IM_H and IM_L are generated with a temporal delay t_{int} that encodes the intensity. Adapted from Liu et al. (2015).

discharged by a current (the photo current). The output voltage is compared to two thresholds (V_H and V_L). In order to reduce comparator mismatch, both comparisons are carried out using a single unit by switching the reference voltage. At the point when a threshold is reached, an AER event is generated. The intensity is then encoded in the temporal delay between the two events generated by the thresholds V_H and V_L. The resulting event rate is three times that of the DVS because a DVS generates one event per detected change, whereas the ATIS generates three events per detected change, two of which are related to intensity measurement. The ATIS uses the concept of time-based intensity encoding, which results in a very high intra-scene dynamic range of 143 dB. However, a significant drawback that comes with this approach is that the intensity measurement may be reset by a new DVS event before it was completed. In particular, this happens in the presence of thin and fast moving objects, which disappear from the output as a result. Such motion artifacts, which are caused by intensity measurements that are not synchronously triggered, make this sensor unsuitable for many machine vision applications. Another drawback of the ATIS sensor is that two photo diodes are used in each pixel.

3.3.3 Other neuromorphic vision sensors

Transient responses, such as the temporal contrast events supplied by the DVS, can be very useful for many tasks such as tracking or localizing fast moving objects. By studying the biological system, it can be observed that the magnocellular system is characterized by transient responses and short latencies and that it is associated with tasks including detection, alerting and visual motion. Thus, this system is often referred to as the “where” pathway. In contrast, the “what” pathway is associated with tasks such as object recognition. The brain has good reasons to separate the “where” from the “what” because different tasks have dif-

ferent requirements. For example, it seems reasonable to expect that the response to "where" something occurs should happen quickly, whereas the question of "what" something is should be answered accurately. Following this analogy, the temporal contrast sensors such as the DVS are suitable for solving "where" questions, but how can "what" questions be addressed? The ATIS sensor was one of the first attempts to provide a dual-pathway output. Another example of a device that provides both a temporal contrast and intensity readout is the dynamic and active pixel vision sensor (DAVIS) (Berner et al., 2013; Brandli et al., 2014a). The DAVIS merges the DVS with a traditional active pixel sensor (APS). The main innovation is that these two sensors are combined at a pixelar level using an extended circuit that involves four additional transistors, whereby the APS and DVS circuit share the same photo diode. This is possible because the DVS pixel does not consume the photo current, and thus, it can be directly used in the APS circuit. The DAVIS has some advantages over the ATIS. Firstly, it provides synchronous intensity frames which are not subject to motion artifacts (this is particularly true of the global shutter form of the DAVIS described in Brandli et al. (2014a)). Furthermore, the rate at which intensity information is output (i.e. the frame rate) can be controlled and tailored to the requirements of the specific application. For this reason, it can be argued that the DAVIS clearly separates the two streams, as is the case in the human brain. A more detailed review of event-based vision sensors can be found in Delbruck et al. (2010b).

3.4 Neuromorphic processors

In its original form, the term "neuromorphic" described analog hardware that exploited the physics of silicon to directly reproduce the biophysical properties of the nervous system. Nowadays, the definition has broadened to include any form of analog, digital or mixed-signal implementation of a neural processing system. Neuromorphic processors have two purposes. In computational neuroscience, they can be used as an alternative to simulations, to investigate the behavior of large-scale spiking neural networks. The gain of such simulations depends on the size of the network and the complexity of the neuronal models. These simulations are often very slow, even when run on powerful computers. Conversely, neural hardware is capable of emulating large-scale neural networks in real-time, regardless of their size. In the field of neuromorphic engineering, neuromorphic processors provide an efficient way to implement event-based computing systems. Many years of research into the brain has revealed that neural dynamics are essential for computation. Thus, the ability to reproduce biologically realistic dynamics is a core requirement for neuromorphic processors. Although from a structural point of view, neuromorphic processors vary greatly, from a function perspective, they all combine many instances of two common building blocks: silicon neurons and synapses.

3.4.1 Neural dynamics in silicon

In order to process signals efficiently and interact with the environment, the time constants which the components of the nervous system are subject to are well matched to the physical processes in the real world. Attempting to produce biologically plausible dynamics in silicon is not a trivial problem, because the carriers of semiconductors (electrons) are orders of magnitude faster than those used by the nervous system (ions). The conductance dynamics of neurons is based on a process whereby ions diffuse across the cell membrane. This occurs at the sites of ion-channels, which are tiny ports in the cell membrane that allow ions to pass through. In general, neuron conductance and synaptic transmission dynamics can be efficiently modeled by first-order differential equations of the form

$$\tau \dot{y} = -y + x \tag{3.5}$$

where y is either a membrane potential or a synaptic current and x is the driving force (Destexhe et al., 1998; Liu et al., 2015). In silicon, the main building block of integrated circuits, the metal-oxide semiconductor field-effect transistor (MOSFET), is also characterized by a diffusion process when operated in the *weak-inversion* or *subthreshold* regime. Here, electrons diffuse from one end (drain) to the other (source), whereby the current exponentially depends on a control voltage applied to the third terminal (gate) of the device. When combined with the integrating mechanism of a capacitor, simple circuits emerge that behave as described by Eq. 3.5. If the state variable y is represented by a physical voltage, the circuit is said to operate in *voltage-mode* (Liu et al., 2002). An alternative approach would be to use *current-mode* circuits, where y is a physical current (Toumazou et al., 1990). In current-mode, first-order differential equations can be efficiently implemented using *log-domain* circuits. These circuits have a couple of advantages such as compactness, low power consumption and a wide dynamic range at low power supply voltages. One way to implement Eq. 3.5 involves the use of a differential pair integrator (DPI) circuit (Bartolozzi et al., 2006; Bartolozzi and Indiveri, 2007). The schematic of the DPI circuit is shown in Fig. 3.4. The differential equation of the circuit can be derived as follows:

$$\tau \frac{d}{dt} I_{out} = -I_{out} + I_{in} \cdot \frac{\frac{I_{out}}{I_{tau}}}{1 + \frac{I_{out}}{I_{thr}}} \tag{3.6}$$

with the time constant $\tau = \frac{CU_T}{\kappa I_{tau}}$. It should be noted that this is a nonlinear first-order differential equation. However, assuming that the input currents are sufficiently large ($I_{in} \gg I_{tau}$), it follows from the steady state solution that $I_{out} \gg I_{thr}$. In this case, Eq. 3.6 can be approximated by a linear first-order differential equation:

$$\tau \frac{d}{dt} I_{out} = -I_{out} + I_{in} \cdot \frac{I_{thr}}{I_{tau}} \tag{3.7}$$

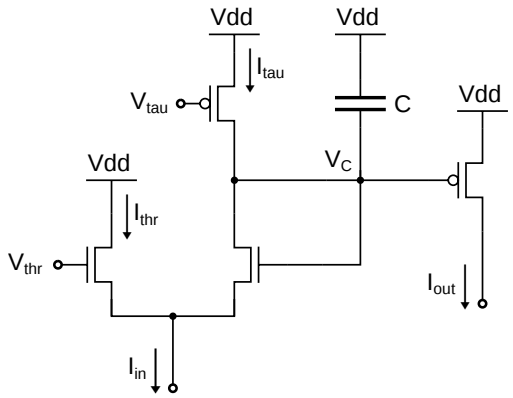


Figure 3.4: Schematic of the differential pair integrator (DPI) circuit.

This equation is in the same form as Eq. 3.5 but in this case, the adjustable gain is proportional to I_{thr}. Although the classic log-domain integrator first proposed by (Frey, 2000) has the same functionality, it requires more transistors and it has fixed unity gain. Another similar log-domain filter called the *Tau-cell* was first described by Edwards and Cauwenberghs (2000) and later put forward as a fundamental building block in the creation of arbitrary differential equations (Van Schaik and Jin, 2003).

3.4.2 The silicon neuron

Silicon neurons exist in various forms, ranging from simple linear-threshold units to complex multi-compartment models. Typically, such models comprise multiple functional blocks which represent the different computational properties of the biological ideal. These include a block to model conductance dynamics, a block to generate spike events, a refractory period block, and a block to adapt spike frequency. The circuits which model conductance dynamics were already discussed in the previous section. Another important function of silicon neurons is to generate spike events, which is usually achieved using a switching amplifier. The membrane potential is fed into the amplifier, which produces a large output, but only after a threshold is reached. This behavior is often achieved by using two inverters in series. The original *Axon-hillock* circuit described by Mead (1989), used the amplified output as a source of positive capacitive feedback that kept the membrane potential above the threshold while the membrane capacitor was discharged. This mechanism allows the silicon neuron to reset itself. An alternative approach is based on current feedback (Culurciello et al., 2003). When adopting this approach, the current of the first inverter is copied and used as positive feedback to accelerate the transition time during which the inverter is in the conducting state. This method not only improves transition speed but also significantly reduces the power consumption of the event generator. The positive feedback which travels through voltage-controlled sodium channels is a well-understood mechanism which depolarizes the action potentials generated in the nervous system. To control the spiking threshold, a transconductance amplifier (Liu et al., 2002) can be used. This amplifier has a bias voltage V_{thr} that determines the switching

point. To reset the membrane potential, the event which is generated usually activates a transistor that discharges the capacitor. The biological analogue of the reset transistor are the potassium channels that initiate the repolarization process. To model the refractory period of neurons, the reset transistor can be connected to a capacitive node that keeps it in the reset state for a short period after the event has been generated. Another characteristic of biological neurons is their ability to adapt the spike-frequency. Neurons tend to reduce their firing rate when stimulated by constant inputs. This mechanism can be implemented using either a negative feedback loop that subtracts an adaptation current from the input (Indiveri, 2007), or an adaptive threshold as described in the neuron model of Mihalas and Niebur (2009).

The so-called "conductance-based silicon neurons" provide an example of a bio-physically realistic implementation of the neuron model. These circuits explicitly model ion-channels by exploiting the similar physics of carrier transport in silicon. Examples of such implementations are found in Mahowald and Douglas (1991); Dupeyron et al. (1996); Simoni et al. (2004); Farquhar and Hasler (2005); Hynna and Boahen (2007). These circuits often require a considerable surface area of silicon and are therefore not suitable for large-scale integration. A good trade-off is achieved using a class of models known as *generalized integrate and fire* neurons (Jolivet et al., 2004). These models have the advantage that they capture many of the computational properties of biological neurons, and can be implemented on compact circuits. An excellent review of existing implementations is found in Indiveri et al. (2011). One of these circuits is called the *DPI neuron*, first proposed in Livi and Indiveri (2009), that implements an *adaptive exponential integrate and fire* neuron (Brette and Gerstner, 2005). The original circuit is illustrated in Fig. 3.5a and comprises four main characteristics. Firstly, the DPI circuit is used to model the leak conductance of the neuron. Secondly, an adaptation block is used to model adaptive conductance in order to adjust the frequency accordingly. Thirdly, an inverting amplifier with positive feedback is used to generate spike-events. Finally, a capacitor is used to control the refractory period. A full description of the circuit and its operation can be found in Livi and Indiveri (2009). An approximate analytical solution was derived in Indiveri et al. (2010), which can be expressed as follows:

$$\begin{aligned} \tau \frac{d}{dt} I_{mem} &= -I_{mem}\left(1 + \frac{I_g}{I_{tau}}\right) + I_{mem_\infty} + f(I_{mem}) \\ \tau_g \frac{d}{dt} I_g &= -I_g + I_{g_{max}} r(t) \end{aligned} \tag{3.8}$$

where I_{mem} is the current-mode representation of the membrane potential and I_g models the adaptive conductance variable. I_{mem_∞} is the steady state solution of I_{mem} in the absence of positive feedback and $I_{g_{max}}$ is the maximum value of the adaptive conductance. The term $r(t)$ is equal to 1 while spike events are being generated and 0 at all other times. The positive feedback is determined by $f(I_{mem})$ which is an exponential function of I_{mem}. Examples of different spiking behaviors achieved with this circuit are shown in Fig. 3.5b.

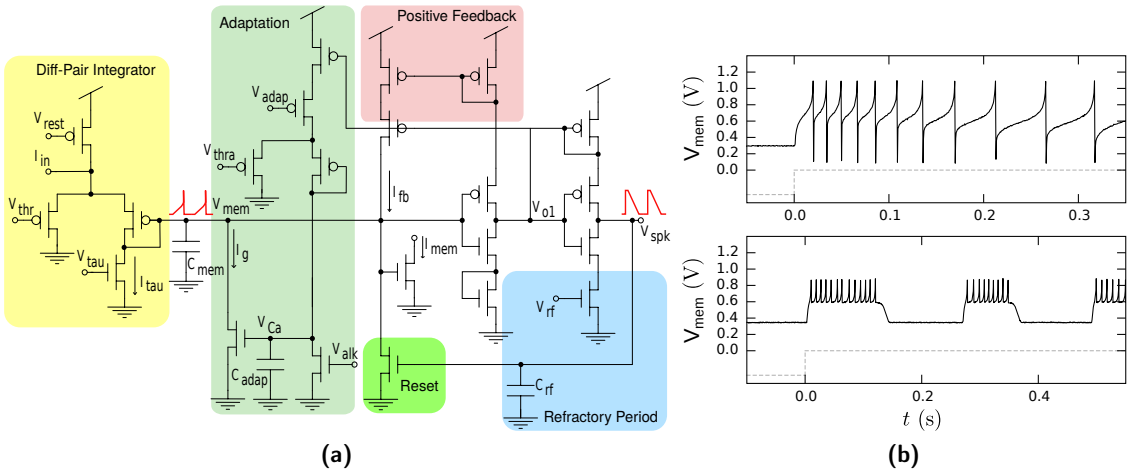


Figure 3.5: The DPI neuron. **(a)** Schematic of the DPI neuron circuit. Adapted from Livi and Indiveri (2009). **(b)** Examples of different neural behavior: Spike-frequency adaptation is shown at the top and bursting behavior is shown at the bottom. The data shown corresponds to real measurements taken from the ROLLS neuromorphic processor (Qiao et al., 2015).

Other types of silicon neurons

The previous section focused on silicon neurons implemented in the form of analog sub-threshold circuits. As mentioned earlier, a variety of other design approaches exist. Examples of silicon neurons that operate in the above-threshold regime are presented in Schemmel et al. (2006) and Wijekoon and Dudek (2008). These circuits operate according to an *accelerated timescale*, however, because the transistors are mostly in *strong inversion*. Alternatively, silicon neurons can also be implemented with the switched-capacitors technique, which uses discrete time and clocked signals (Serrano-Gotarredona et al., 2006; Folowosele et al., 2009; Mihalas and Niebur, 2009). More recent designs envisage fully digital silicon neurons that use digital adders and accumulators instead of capacitors. Such designs were presented in Camunas-Mesa et al. (2008) and Merolla et al. (2011). A detailed review and comparison of all approaches can be found in Indiveri et al. (2011).

3.4.3 The silicon synapse

Synapses form the connections which transfer information among neurons. The nervous system engages two kinds of synapses, that differ in the way in which conduction is passed from one cell to another. While *electrical synapses* are simple physical links, *chemical synapses* form a junction between two cells. In the latter, the signal transmission is based on a chemical process that involves a complex molecular machinery. The vast majority of synapses in the brain are chemical. Chemical synapses are superior to electrical synapses in the way they employ properties such as amplification, dynamics and plasticity, all of which significantly aid neural computation. Therefore, silicon synapses attempt to reproduce the behavior of chemical synapses.

Synaptic dynamics in silicon

The dynamics of synaptic transmission can also be efficiently modeled using first-order differential equations of the same form as Eq. 3.5 (Destexhe et al., 1998). In this regard, the driving force x is a spike, often in the form of a short pulse, and the state variable y is a post-synaptic current (PSC). In the nervous system, PSCs typically exhibit a charge phase, during which the current increases, followed by a discharge phase, during which it exponentially decays to the reset level. Such behavior is achieved using a first-order differential equation, whereby the charge phase is active during the pulse and the discharge phase begins directly after the pulse. Early implementations of this model are described in Lazzaro et al. (1994), Boahen (1997) and Arthur and Boahen (2004). All of these implementations share a common drawback in that they do not sum the effects of multiple subsequent spikes linearly. Linear summation of PSCs is a desired property of many neural systems. Once again, however, current-mode, log-domain circuits can provide a solution to this problem. The log-domain synapse (Merolla and Boahen, 2003) and the DPI synapse (Bartolozzi and Indiveri, 2007) both sum linearly. This has the advantage that the same silicon circuit can be used to sum the contributions of spikes from different sources. Particularly when large-scale integration is required, this can save a significant amount of silicon real estate. The circuit of the DPI synapse is equivalent to the DPI integrator (Fig. 3.4) with an input $I_{in} = I_w$ that reflects the synaptic weight. Thus, the circuit's differential equation yields

$$\tau_{syn} \frac{d}{dt} I_{syn} = -I_{syn} + I_{thr} \cdot \frac{I_w}{I_{tau}} \tag{3.9}$$

Once again, the adjustable gain I_{thr} proves beneficial compared to alternative implementations (such as the log-domain synapse) which have a fixed gain. When the gain is fixed, it is often too small in order to observe the effect of a single spike on the membrane potential of the post-synaptic neuron. Biologically more realistic implementations of synaptic dynamics are known as *conductance-based* circuits. Such circuits incorporate the dependence of the post-synaptic current on the membrane potential. Implementations that consider conductance-based dependencies are described in Farquhar and Hasler (2005), Hynna and Boahen (2007) and Vogelstein et al. (2007).

Synaptic plasticity in silicon

Probably the most important feature of the nervous system is its ability to dynamically change synaptic efficacy. This mechanism, commonly termed *synaptic plasticity*, is a key ingredient in the process of learning. Synaptic plasticity occurs on different time scales. Short-term plasticity (STP) is solely driven by pre-synaptic activity and it has short time constants, ranging from milliseconds to seconds. STP describes a form of temporal filtering which has useful computational properties (Fortune and Rose, 2001). There are two forms of STP: *depression* and *facilitation*. In the case of depression, the effect of consecutive spikes is gradually reduced,

whereas the process of facilitation denotes the opposite. Both processes can be modeled as linear filters with exponential decay (Varela et al., 1997; Maass and Sontag, 2000) and both can be implemented in silicon using similar circuits to those previously described (Rasche and Hahnloser, 2001; Boegerhausen et al., 2003).

While STP has useful computational properties, long-term plasticity (LTP) is the essential process that makes it possible for a neural network to learn a task and express behavior. LTP occurs on a time scale ranging from minutes to hours, days or even years. Unlike STP, the process is driven by both pre-synaptic and post-synaptic activity. Models of learning have been studied extensively and various learning rules have been proposed, ranging from the abstract to the biologically realistic. The implementation of LTP in silicon is not a trivial challenge, for many reasons. On the one hand, the long time constants that are required for LTP are very difficult to realize in silicon, mainly due to the effect of leakage currents. On the other hand, learning circuits are often complex and expend serious quantities of silicon real estate. Last but not least, the implementation of LTP is often limited to a very specific learning rule, which inevitably imposes constraints on the type of task that can be potentially performed using the neural hardware. For these reasons, neuromorphic processors often employ programmable synapses, whereby the learning rule is implemented off-chip. However, considerable focus is now being placed on implementing the learning rules directly in silicon. Due to the spiking nature of neuromorphic hardware, such implementations are typically spike-based and often use the concept of *bistable* synapses in order to deal with the problem of leakage (Fusi et al., 2000; Indiveri, 2002; Indiveri et al., 2006; Brader et al., 2007). A more detailed review of different types of dynamic silicon synapses can be found in Liu et al. (2015).

3.4.4 Large-scale systems

Biological systems employ billions of neurons to produce complex behavior. Since neuromorphic hardware aims to solve similar tasks, scalability becomes a very important aspect. The main challenge is that biology has built the brain in three dimensions while, thus far, integrated circuits have to cope with a two-dimensional substrate. Although multi-chip setups can be used to emulate up to millions of neurons, communicating among them tends to pose a difficult challenge. This often limits scalability or imposes constraints on the network topology. Therefore, building *large-scale* neuromorphic systems requires substantial investment in research and engineering, in order to design the architecture and develop the communication systems necessary. In the following subsection, the emerging, state-of-the-art approaches to building large-scale systems are briefly summarized.

The Neurogrid project at the University of Stanford integrates 16 Neurocore chips that emulate a million neurons with billions of synapses in real-time. This is all implemented on a board that consumes only three watts (Benjamin et al., 2014). A Neurocore chip is a full-custom ASIC that contains an array of 256×256 neurons, implemented with analog circuits. Each chip is connected to three others and the routing network is organized in a tree structure. A team at

the University of California in San Diego has taken a different approach, proposing a modular, hierarchical architecture that combines an integrate-and-fire array transceiver (IFAT) (Yu et al., 2012) with 65k switched-capacitor analog neurons and routing fabrics contained in each single node (Joshi et al., 2010). These nodes are connected in a linear array. A second-level router at the end of this array routes local spikes within it. If a spike has no target in the local routing table, it is forwarded to a third-level router that handles communication among arrays of nodes. This scheme can be extended to an arbitrary number of levels. A completely different approach has been adopted by the University of Heidelberg in the BrainScaleS project (former known as the FACETS project). This approach integrates the cores of analog silicon neurons and synapses on a waver scale (Schemmel et al., 2008). The communication among the neurons is carried out through multiple parallel lines that run horizontally and vertically between the cores, whereby the connectivity is programmed with static switches. This approach is significantly different to other neuromorphic systems in that it enables accelerated emulation of up to 100,000 times faster than real-time. Yet another, very recent approach, is IBM's TrueNorth project. One million neurons and 256 million configurable synapses have been integrated on a single chip that consumes only 63 milliwatts (Merolla et al., 2014). The design is fully digital and it applies an efficient combination of asynchronous and synchronous logic. A sophisticated communication interface allows the chips to be tiled in two dimensions in order to form neural networks of arbitrary size. The TrueNorth approach is very different from other neuromorphic systems in that it is completely deterministic. This enables simulation and makes it easier to design neural systems. Finally, the SpiNNaker project at the University of Manchester aims to build a massively parallel computer that simulates up to a billion neurons in real-time (Furber et al., 2014). The SpiNNaker system is probably the most advanced *general-purpose* large-scale neuromorphic system. The core element is the SpiNNaker chip, a semi-custom designed ASIC. This chip integrates 18 ARM processors used for neural computation together with a dedicated asynchronous router that handles communication among the chips. On a single ARM processor, about one thousand neurons can be simulated in real-time. Each SpiNNaker chip is connected to six others, forming a two-dimensional torus network which allows arbitrary scalability. One of the main advantages of this approach is that the use of general-purpose processors to simulate neural behavior allows the implementation of arbitrary models.

3.5 Event-based machine vision

A substantial proportion of the human brain is devoted to processing visual information. It seems obvious that by understanding the key concepts related to how the brain functions, an improved approach to machine vision can be developed. The novel field of *event-based machine vision* applies the principle of self-timed (or event-based) computation to machine vision by using neuromorphic vision sensors. Since the first easily accessible sensors were temporal contrast sensors such as the DVS or ATIS, the field has been mainly focused on developing algorithms for temporal contrast events. Some of the obvious, inherent advantages

of the sensor include the low latency, absence of redundancy and high dynamic range. Furthermore, the data produced by these sensors has some interesting properties which can be exploited to solve some difficult problems in the field of machine vision. For example, optical flow can be inferred directly from the temporal sequence of events (Benosman et al., 2014; Tschechne et al., 2014). Another example relates to the visual correspondence problem, which has proven to be a great challenge in the case of many machine vision applications. This will be the main subject of the next chapter.

In general, the main challenge of event-based machine vision concerns how to handle time. Unlike classical image frames, the visual data is not synchronized at fixed time intervals. Thus, time has to be dealt with differently. A trivial method would involve accumulating events over time in order to generate frames that can be processed subsequently in the classical frame-based manner. While this could be more efficient than the traditional approach (with frame-based cameras) for certain applications, this would be merely a consequence of the superior sensor rather than the algorithm (which is the same). Nevertheless, this simple approach has the additional advantage that the "exposure time" of the generated frames can be chosen arbitrarily. Thus, motion blur caused by fast moving objects can be prevented. Frames that are generated based on a temporal criterion will inevitably depend on the velocity present in the scene context. To overcome this, frames can be generated with a fixed number of events. In this case, the frame-rate automatically adapts to the velocity of the objects in the scene. Even if the frames are generated in this way, however, the process is still a form of synchronization. The full potential of event-based computation will only be reached if the data is processed in a completely *frameless* and *asynchronous* manner. This means that every individual event is processed at the time it occurs and the believed current status is updated accordingly. This idea is better understood by looking at a simple example. Consider the task of tracking a target, given its initial position. A very simple, yet robust solution would be to iteratively "push" the estimated position of the target in the direction of the location of new events. As the position of the target is updated after every event, this method outperforms any frame-based approach. The obvious benefit illustrated in this example raises the question of why the event-based approach has not yet outflanked frame-based approaches to machine vision. This is mainly for two reasons. Firstly, while the stated example involves a straightforward rule to update the believed current position of the target, this is often not so trivial in the case of more complex problems. Machine vision is a field that has evolved over many years to become what it is today, and since existing algorithms cannot be directly applied to event-based data, event-based algorithms have to be developed first. Secondly, even after a theory has been formulated, the computation required for each event often results in systems with slow performance. This is mainly because today's hardware is optimized for processing synchronous data, particularly in the form of images. Thus, it is often very difficult to execute event-based algorithms in a massively parallel manner on normal hardware. The constraints of this hardware mean that such algorithms must be chiefly executed in a serial manner, which leads to the obvious drawback of slow performance. For this reason, the success of event-based machine vision is not only a question of theory and algorithms,

but also a question of advances in hardware. Indeed, neuromorphic processors are being developed with the aim of solving this exact problem.

A complete overview of the field of event-based machine vision is given in Brändli (2015). The author has divided the field into four main categories: localization, identification, reconstruction and feature extraction. Most of the event-based algorithms that have been proposed so far concern localization. This includes various forms of tracking such as convolution-based tracking (Serrano-Gotarredona et al., 2009), geometrically constrained tracking (Litzenberger et al., 2007), event-cluster tracking (Cardinale, 2006; Drazen et al., 2011), shape-based tracking (Ni et al., 2012) or active-marker tracking (Muller and Conradt, 2011; Censi et al., 2013). Also included and closely related to tracking is the estimation of the optical flow (Benosman et al., 2014; Tschechne et al., 2014), self-motion tracking in the form of visual odometry (Censi and Scaramuzza, 2014; Mueggler et al., 2014) or simultaneous localization and mapping (SLAM) (Weikersdorfer et al., 2013). Finally, the estimation of depth is carried out using another form of event-based localization algorithm, which comprises two classes of approaches. *Active* depth estimation usually involves a source of structured light and an event-based camera (Brandli et al., 2014b). The *passive* approach to depth estimation is *event-based stereo vision*, which is the main subject of this thesis.

3.5.1 Event-based stereo vision

Event-based stereo vision commonly describes many different approaches involving both hardware and software. Tab. 3.1 sorts the existing research into two categories, either hardware or software, depending on how it is implemented. In addition, the approaches are categorized according to whether the hardware is analog, digital or mixed. In most, but not all cases, the research related to hardware has aimed to build a complete event-based stereo vision system comprising sensing and processing units. The research related to software mainly involved developing theory and algorithms. Event-based stereo vision systems are covered in the next section, while all other work is discussed in the relevant sections of the thesis, as indicated in the table.

3.6 Event-based stereo vision systems

The pioneering work of Mahowald and Delbrück (1989) led to the first neuromorphic stereo vision system implemented on an analog VLSI system (Mahowald, 1994b). Unlike today's event-based approaches which process digital events, this system used time-continuous analog signals to encode the spatial contrast generated by the adaptive photoreceptors of the first silicon retina (Mahowald and Douglas, 1991). The system employed a cooperative network of analog units integrated on a VLSI chip. The network implemented the algorithm of Marr and Poggio (1976) and the system remained continuously responsive to changes in the retinal input. Thus, the computation was driven by the data, which was an early form of event-based processing. The work of Mahowald was not only a pioneering contribution to the field of

Table 3.1: Categorization of event-based stereo vision. Hardware is subdivided into analog (A), digital (D) and mixed (M).

Approach	Category	Short description	Section
Mahowald (1994a)	HW (A)	Analog VLSI chip for stereocorrespondence	3.6
Tsang and Shi (2004)	HW (M)	Neuromorphic implementation of the binocular energy model	3.6
Shimonomura et al. (2008)	HW (M)	A multichip system that emulates disparity computation in V1	3.6
Schraml et al. (2010b)	HW (D)	Dynamic stereo vision system for real-time tracking on a DSP	3.6
Eibensteiner et al. (2012)	HW (D)	Event-based stereo vision algorithm for implementation on a FPGA	3.6/4.2.1
Belbachir et al. (2014)	HW (D)	Real-time panoramic stereo vision with event-based line sensors	3.6
Camunas-Mesa et al. (2014)	HW (D)	Event-driven stereo vision based on FPGA convolution boards	3.6
Hess (2006)	SW	Global and local disparity filters based on temporal correlation of events	4.2.1/5.1.3
Stephan Schraml (2007)	SW	Area-based stereo vision based on generated frames from accumulated events	4.2.1
Kogler et al. (2011)	SW	Time-based approach that makes explicit use of event timing	4.2.1
Rogister et al. (2012)	SW	Event-based stereo matching with epipolar constraint	4.2.1
Carneiro et al. (2013)	SW	Event-based N-ocular 3D reconstruction based on epipolar constraints	4.2.1
Piatkowska et al. (2014)	SW	Asynchronous stereo vision based on a cooperative network	5.1.3
Firouzi and Conradt (2015)	SW	Asynchronous stereo vision based on a cooperative network	5.1.3
Mueggler et al. (2015)	SW	Stereo matching of event-based circle trackers	3.6

neuromorphic engineering, it also provided insight into the biological process of stereopsis. More recent work has involved building neuromorphic stereo vision systems with the goal of implementing the exact biological models of stereopsis. Tsang and Shi (2004) presented a neuromorphic implementation of a disparity-tuned complex cell which combined event-based spatiotemporal contrast sensors (Zaghloul et al., 2004a,b) with chips which carried out orientation filtering (Choi et al., 2004). Indeed, the results of experiments using this setup have enabled researchers to predict the role of specific disparity cells in the visual cortex. A

similar neuromorphic implementation of the disparity energy model comprised the silicon retina of Kameda and Yagi (2003) and the orientation chips of Shimonomura and Yagi (2005) together with an FPGA (Shimonomura et al., 2008).

The introduction of the DVS, however, marked the real beginning of the development of event-based stereo vision systems. Since then, the overwhelming majority of all proposed stereo vision systems have been based on processing temporal contrast events. Embedded stereo vision systems, incorporating two DVS sensors and a digital signal processor (DSP), were built to enable a low-cost pre-crash warning system for cars (Kogler et al., 2009) and a system to track pedestrians in real-time (Schraml et al., 2010b,a). An improved dynamic stereo vision system was built comprising two ATIS sensors, an FPGA and a DSP which could detect if an elderly person falls at home (Belbachir et al., 2012). Eibensteiner et al. (2014) proposed a high-performance architecture for event-based stereo vision, implemented on a FPGA platform that provides depth maps with an equivalent frame-rate of up to 1140 Fps. Belbachir et al. (2014) used a pair of rotating line DVS sensors together with two low-cost FPGA boards to provide panoramic 360° depth maps at up to 10 rotations per second. Finally, an event-based stereo vision system comprising a stereo DVS rig and a smartphone processor was mounted on a quadrotor to enable it to avoid obstacles (Mueggler et al., 2015). The underlying algorithms of these systems are explained in Section 4.2.1.

3.7 Discussion

Neuromorphic engineering has emerged from the idea of replicating the organizing principles of the human brain in silicon. Initially, optimists predicted that neuromorphic systems would surpass traditional computers in performing human tasks. To date, however, this prediction has not come to pass — but why not? When it comes to understanding large-scale neural networks, the last decades of studying neural computation have shown that it is not sufficient to simply understand the biophysical mechanisms of the nervous system. Most of these mechanisms are quite well understood in detail, but they just form a set of rules with a vast amount of parameters. In order to *build* neural networks, however, researchers need to determine which parameters are critical and how they should be set. While nature has spent millions of year exploring this parameter space, we are merely at the very beginning of the search. In this respect, it seems premature to try to build neuromorphic hardware that models the realistic biophysical mechanisms of the nervous system without first understanding how these mechanisms contribute to the intelligent behavior of a large-scale neural network. Over the last years, neuromorphic engineering has increasingly evolved towards becoming a pure engineering discipline. For this reason, while much research has aimed to create large-scale networks in silicon, the question of how these networks should be used has been left for others to debate. Similarly, the silicon retina also evolved into an increasingly efficient machine vision sensor, which served as a catalyst to the new field of event-based machine vision. At the beginning, some of the applications of event-based machine vision demonstrated its benefits and clearly outperformed traditional approaches in carrying out simple tasks. These

demonstrations, however, simply exploited the efficiency of the modern silicon retina but did not involve new algorithms. Instead, data was processed in the traditional frame-based manner. Initially, therefore, the field diverged quite far from its purpose of initiating a shift in machine vision to a computing paradigm that is inspired by human vision. Such simple demonstrations, however, did not draw enough attention from the machine vision community. Furthermore, the rapid improvement of vision sensors and processors has put increasing pressure on researchers to translate the conceptual advantage of event-based systems into practical gains. This has led to the development of novel algorithms that exploit the full potential of event-based computation. Recent work has shown that while these algorithms work quite well in theory, they are inefficient when implemented on traditional hardware. Thus, neuromorphic processors provide a promising alternative. For this reason, the software for neuromorphic hardware is likely to begin to become available and it will finally be established whether neuromorphic engineering can live up to its promise.

4 A Novel Approach to Event-based Stereo Vision

4.1 Space-time representation of visual information

In 1878, when Muybridge came up with the proof that at some instants, a horse has all four hooves off the ground, he laid the foundations of today's camera technology. He succeeded in doing so using a row of cameras with interconnected shutters which were sequentially triggered, providing a series of stroboscopic images. This way of visually capturing a dynamic scene was one of the first of its kind and is a perfect example for what can be termed "the space-time representation of visual information". His technique, which involves synchronized capturing of static images at discrete points in time, defined this method of space-time sampling. It might be expected that after more than a 100 years of technological progress, more sophisticated sampling techniques would have evolved. Surprisingly, however, the whole field of machine vision still exclusively relies on this method. An obvious, related drawback is that redundant information associated with each image needs to be processed at places where the content has not changed. This becomes particularly serious when high temporal precision (i.e. high frame rates) is required, as is often the case in many machine-vision applications. Nature copes with the problem of redundancy by applying a more sophisticated method of space-invariant and asynchronous space-time sampling (Polimeni and Schwartz, 2001). The term "space-invariant" refers to a non-uniform distribution of photoreceptors, characterized by a higher density in the center (fovea) and a decreasing density towards the periphery. "Asynchronous" describes the temporal aspect of data-driven and self-timed sensing.

4.1.1 Space-time sampling strategies

The following theoretical considerations provide a deeper insight into the effect of space-time sampling. Firstly, consider a one-dimensional world from which generalizations can be drawn. A dynamic scene can be described by an intensity function in space-time $I(x,t)$. Now, assume the classical sampling method is applied with a fixed spatial and temporal sampling rate of K and Ω respectively. The upper limit on measurable resolution is determined by a sinusoidal

scene component:

$$I_C(x,t) = \mathrm{e}^{iKx}\mathrm{e}^{i\Omega t} = \mathrm{e}^{iK(x+\frac{\Omega}{K}t)} \tag{4.1}$$

which corresponds to a feature of spatial frequency K moving at a velocity $v = \frac{\Omega}{K}$. Thus, when sampling at a spatial frequency of K and a temporal frequency of Ω, the measurable velocity will have an upper limit of $\frac{\Omega}{K}$ (Polimeni and Schwartz, 2001). This formula has been confirmed through experiments carried out by Polimeni and Schwartz (2001). The authors found that for a standard 30 Hz, 512×512 pixel camera, a velocity of about 1000 pixel/s was the maximum speed at which they could accurately track an object containing features with a spatial frequency of K. This indicates that an object is moving across the image at a surprisingly slow speed within half a second. Hence, when spatiotemporal sampling issues are fully considered, it becomes clear that the classical synchronous sampling strategy entails strict limitations on all areas of performance related to visual motion (e.g. motion tracking). Indeed, this might be one of the reasons why nature evolved an asynchronous sampling strategy.

Consider a hypothetical (artificial or biological) visual sensor with a spatial resolution of n^2 and a maximum sampling rate of Π (samples per second). In what way will the sampling strategy which is adopted affect the temporal resolution of this sensor? If a classical synchronous sampling method is used, similar to that of a video camera, the temporal resolution would be limited by a fixed temporal sampling frequency (frame-rate) $\Omega_s = \frac{\Pi}{n^2}$. In contrast, the asynchronous model only samples when the temporal intensity changes, meaning that no redundant information is captured. Assume that r is a metric measuring the average quantity of redundant information which ranges from 0 to 1. If $r = 0$, the intensity is constantly changing at every spatial location of the scene. Conversely, if $r = 1$, there is no fluctuation in intensity at all. In this scenario, the frequency $\Omega_a = \frac{\Pi}{n^2(1-r)}$ can provide an estimate of the average temporal resolution which an asynchronous sampling model can achieve. Unfortunately, r is difficult to measure because it depends on the scene context and scene dynamics. However, it is evident that an inequality of $\Omega_a \geq \Omega_s$ always holds. This means that at worst, both methods have equal temporal resolution, but for literally any realistic scene the asynchronous sampling method performs much better. For example, consider a fly maneuvering within the field of view of the sensor at a constant distance, so that its projection on the image plane is exactly one pixel in size. The intensity can only change at the fly's current position at each moment in time. Thus, the redundancy is equal to $r = 1 - \frac{1}{n^2}$. By comparing the frequencies Ω_a and Ω_s, an impressive gain of n^2 can be observed[1]. Assuming a standard 512×512 pixel resolution, this would imply that the asynchronous model can achieve a temporal resolution which is superior by a factor of 250,000. Although this is an extreme scenario, this example illustrates the immense potential benefit of asynchronous sampling. Thus far, the lower and upper boundaries in two extreme scenarios have been examined. Can a statement be made

[1] $\frac{\Omega_a}{\Omega_s} = \frac{\Pi}{n^2(1-r)}\frac{n^2}{\Pi} = \frac{1}{1-r} = \frac{1}{1-(1-\frac{1}{n^2})} = n^2$

about more common scenes, however? The redundancy of a scene is very context dependent, meaning that it is hard to find a general expression. Nevertheless, the assumption can be made which supposes that intensity changes are solely caused by movement of the spatial intensity gradients (edges), which implies no changes in scene illumination. In this case, an asynchronous sensor would only sample at a given location if the edges are moving. By further assuming that all edges move with the velocity of the fastest edge, the redundancy can be approximately defined as $r = 1 - \rho$, where ρ denotes the density of the moving edges. The density is obtained by dividing the accumulated length of the moving edges by the total image area. Again, a problem stems from the fact that ρ is strongly dependent on the scene context. However, it is possible to examine how it scales in proportion to spatial resolution. In general, it might be expected that edge density would remain constant as spatial resolution increases. Fortunately, this worst case scenario is very unlikely as it would imply that the whole scene is covered by some sort of line fractal structure. This is the case in certain undersampled parts of the scene, where further edge details are revealed upon zooming in (e.g. the letters of a book title that suddenly became readable at an adequate resolution). Conversely, the edge density in parts of the scene that are oversampled decays with n^{-1}. This leads to the following general relationship:

$$\rho \propto n^{-\beta} \quad \beta \in [0,1] \tag{4.2}$$

In most practical cases, however, scenes tend to be oversampled as the resolution is usually chosen to ensure that a certain feature (only present at specific locations) can be precisely located. This suggests that for many realistic scenes, β is close to 1. Given the fact that the asynchronous sensor only captures moving edges, which are a fraction of the total number of edges in the scene, β approaches even closer to 1. Under the stated assumptions, a relationship can now be derived to model the asynchronous sampling frequency:

$$\Omega_a \propto \frac{\Pi}{n^{2-\beta}} \tag{4.3}$$

While this relationship does not provide information about the absolute temporal sampling frequency, it can yield valuable insight concerning its scalability. This is particularly important in a scenario where a higher spatial resolution is required in order to more precisely locate features, rather than detecting smaller features. In such a case, the assumption $\beta \approx 1$ holds and Ω_a is then found to be proportional to $\frac{\Pi}{n}$. Previously in this section, the conclusion was reached that an asynchronous sampling strategy always outperforms a synchronous one. In addition, it can now also be concluded that asynchronous sampling generally scales better. For example, consider a scenario where an object should be reliably tracked. If a synchronous sampling strategy is used and the precision of spatial localization is doubled, this leads to a fourfold reduction in temporal resolution. Conversely, if an asynchronous sampling strategy is adopted, temporal resolution is only reduced by a factor of two. Another application that

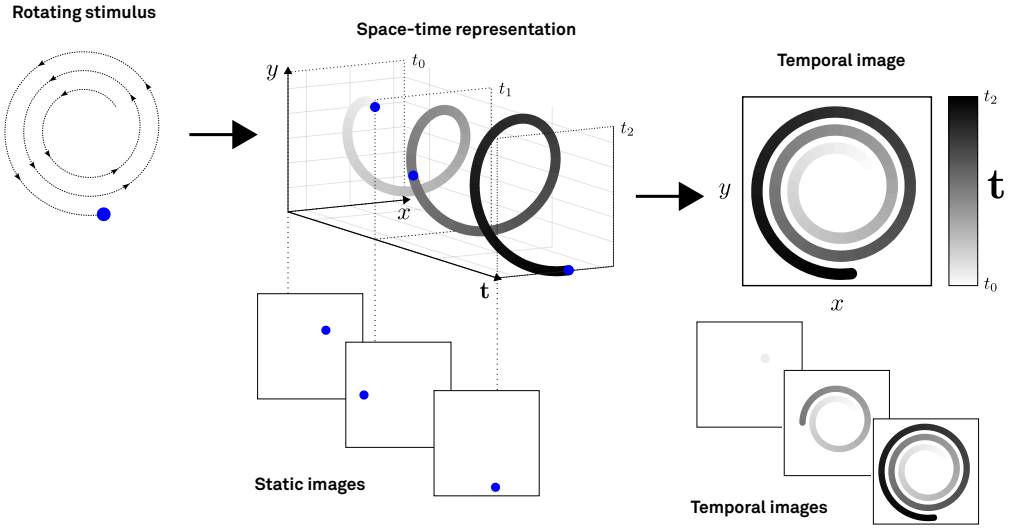


Figure 4.1: The principle of temporal images. Top: A rotating stimulus is shown on the left and its space-time representation in the middle. The resulting temporal image is shown on the right. Grey values encode time. Bottom: Static and temporal images of the same scene at different points in time.

strongly benefits from this relaxed scaling law is stereo vision, where the depth error is a function of spatial resolution.

4.1.2 Temporal image representation

In the following subsections, an intuitive representation of asynchronously sampled visual data is introduced. For the sake of clarity, these are termed *temporal images* as opposed to the conventional *static images* usually obtained from synchronous visual sensors. The concept is best explained by the illustration in Fig. 4.1. Consider an object (blue dot) rotating around a fixed midpoint such that the radius continuously increases. The respective space-time representation of this dynamic scene is depicted by the helix in the middle plot. Incidentally, the levels of grey here do not correspond to intensities but to the time at which the intensity changes (asynchronous samples). At given points in time t_0, t_1 and t_2, the blue dot is overlaid. The corresponding static images are shown below. Throughout this project, absolute image intensities are not used, meaning that color scales can be repurposed to encode sampling times instead. This principle is illustrated on the right-hand side of the figure and informs the definition of temporal images. The temporal images at times t_0, t_1 and t_2 are shown below as analogous to the static images. Note how the temporal images contain all information regarding space-time structure, but not the absolute intensity values. In addition, temporal images can only hold one value for each spatial position, which might lead to a loss of information at places where old samples are overwritten by new ones.

4.1.3 The optimal vision sensor

Given that a hypothetical visual sensor is solely limited by its sampling rate, the ideal implementation of that sensor would apply an asynchronous sampling strategy. Therefore, two mechanisms are required: one to detect changes in intensity and one to measure intensities. The measure of intensity could either be absolute or relative. However, in practice, the latter would suffer from drift error because the samples would have to be integrated in order to obtain absolute values. Inspired by the human eye, a variety of retinomorphic, event-based visual sensors (Posch et al., 2014) have been developed. These sensors aim to exploit asynchronous sampling strategies in order to overcome the bandwidth and latency limitations of conventional cameras (see Section 3.3). To date, the design that probably most closely equates to the aforementioned concept of an ideal visual sensor is the asynchronous time-based image sensor (ATIS) (Posch et al., 2010b). Each pixel comprises a circuit which detects intensity changes. This circuit triggers an absolute intensity readout when a certain threshold is reached. In addition, these pixels are not governed by global control signals, but determine and distribute information autonomously. The core technology in this sensor was already applied in a previous design, the dynamic vision sensor (DVS) (Lichtsteiner et al., 2008), although it could not measure absolute intensities. This functionality has been added in a more recent iteration, the dynamic and active pixel vision sensor (DAVIS) (Berner et al., 2013). The DAVIS differs to the ATIS, however, in that it captures absolute intensities in a frame-based manner at low rates, similarly to conventional CMOS imagers. While this method deviates from the theoretically ideal sampling strategy, unlike the ATIS, it is not subject to motion artifacts and thus, it is more suitable for machine-vision applications (see Section 3.3 for a comparison and further details). Within the scope of this project, all three sensors were used in various experiments but none of the algorithms made use of absolute intensities. Therefore, all three sensors can be considered identical except for their difference in terms of spatial resolution.

4.1.4 The effect of asynchronous space-time sampling on the stereo correspondence problem

In order to compute binocular disparity in the brain, the prevailing key principle underlying the process is based on local cross-correlation models (Filippini and Banks, 2009). This approach has been successfully adopted by many stereo algorithms in machine vision. These methods usually only consider static images as obtained when a synchronous sampling strategy is adopted. Local image patches from different visual sources are then correlated in order to find commonalities. The problem with this approach is that image details (high-frequency content) located at different depths result in weak correlation results (Marr, 1982). Fig. 4.2 illustrates this effect; a stereo scenario is shown, in which two moving stimuli are captured by two one-dimensional sensors. If assumed that the image acquisition time is short (thus negating motion blur), the images captured by conventional cameras would look similar to those in Fig. 4.2b. Since the stimuli are located at different depths, the projections of their images are diversely spaced. Consequently, the correlation of the image signals f_L and f_R produces four

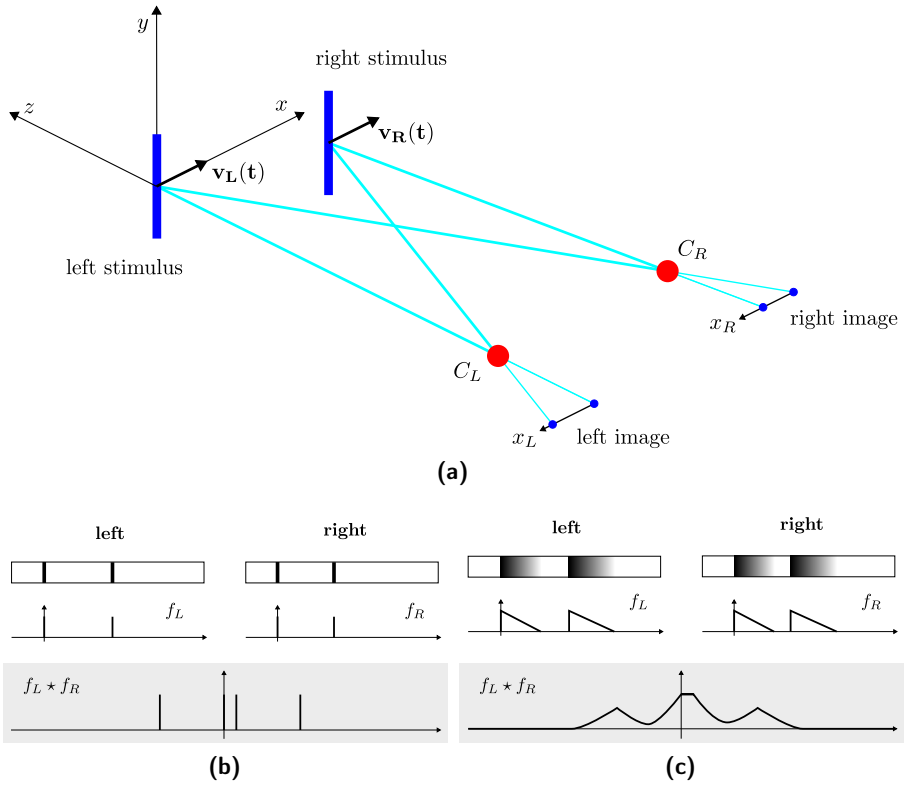


Figure 4.2: Stereo from correlation of static and temporal images. **(a)** A typical stereo scene comprising two moving stimuli and two one-dimensional sensors centered at C_L and C_R respectively. **(b)** Static images from synchronous sampling of the above scene and cross-correlation of the static image signals. **(c)** Temporal images from asynchronous sampling of the above scene containing spatiotemporal contrast. The image signals are a function of space and time and therefore also contain information of preceding stimuli positions which benefits the cross-correlation.

equal peaks without providing a solution to the correspondence problem. This shortcoming is typically circumvented by considering low-frequency content, which could be obtained by low-pass filtering the image signals, for example. It has been shown that the ideal correlation window area is inversely proportional to the spatial image frequency (Banks et al., 2004, 2005). Thus, in order to estimate coarse scene disparity, large correlation areas with low-frequency content are required. For this reason, many algorithms incorporate an incremental approach, starting with coarse matching of low-frequency content and gradually moving towards image details. Instead of increasing the complexity of the matching process, this paper demonstrates that an alternative solution is possible if temporal scene dynamics are considered. This idea is illustrated in Fig. 4.2c. Here, the image signal is a temporal image. Thus it is not only a function of space but also time, meaning that it encodes spatiotemporal information. Such temporal images contain a history of the preceding positions of the stimuli, which gives rise to a correlation signal that now shows dominant peaks at the positions corresponding to the correct disparities of the stimuli. The main advantage of temporal image signals is that they contain low-frequency (obtained from motion) and high-frequency content (obtained from

spatial context) simultaneously. If temporal scene dynamics are considered, therefore, it is reasonable to expect that a simple, cross-correlation approach in which a fixed window size is adopted is sufficient for reliable stereo matching.

4.2 Event-based stereo matching based on local spatiotemporal correlation

The pioneering work of Marr and Poggio (1976), Mahowald and Delbrück (1989) and Mahowald (1994a) laid the foundations for event-based stereo vision more than 20 years ago. With recent advances in neuromorphic engineering, interest in event-based stereo vision has increased and quite a few algorithms have been proposed. Interestingly, these algorithms are to a large extent very closely related to the main concepts of the historical work. Although they are described as new or innovative, in fact, they are simply a reinvention of long-existing ideas. For the present study, a rigorous analysis of the commonalities between these ideas has been undertaken, in order to condense them into simple, yet essential, key principles of event-based stereo vision. Three key principles were identified, which are very closely linked to those contained in the pioneering work cited above but formulated in a more general way:

1. **Aggregation of disparity evidence**: Stereo algorithms inevitably produce noisy outputs due to ambiguity, but by accumulating evidence of equal disparity, this ambiguity can be resolved. This principle is based on the idea of the "cohesiveness of matter" and led to the famous "mechanism of cooperation" (Marr and Poggio, 1976).

2. **Suppression of ambiguity**: Physical limitations impose a constraint on uniqueness, meaning that any point in one view can only correspond to one point in another view. The idea of a uniqueness constraint was also first introduced in Marr and Poggio (1976). This led to the concept of mutual inhibition and the proposal of a winner-take-all mechanism to weight evidence of differing disparity.

3. **Temporal correlation**: The temporal correlation of occurrences in both views (a) and the accumulation of such evidence over time (b) can benefit the matching process significantly. This principle originates from the work of Mahowald and Delbrück (1989) and Mahowald (1994a). It was introduced with the invention of the silicon retina.

In the following section, the existing literature on event-based stereo vision is reviewed. It is argued that all of this research is linked to the three key principles stated above.

4.2.1 Related work

A complete list and categorization of existing work on event-based stereo vision was provided in Section 3.5.1. While complete event-based stereo vision systems have already been

examined in Section 3.6, the focus here is on how the underlying algorithms relate to the three key principles outlined above in (1), (2) and (3a/b). The first work that used the new generation of silicon retinas (here the DVS) is described in Hess (2006), whereby single events were correlated over time (3a) and the results were accumulated in a disparity matrix representing the 3D space. Disparity filters, which apply both local and global smoothing in disparity space (1), were investigated. This is the earliest piece of research examined during the course of this study that makes explicit use of the precise timing information of single events. Other contemporaneous research, such as that undertaken by Stephan Schraml (2007), tested classical area-based stereo vision algorithms such as SAD and SSD, on grayscale images which were generated by accumulating events from silicon retinas over time. A feature-based approach, whereby features were extracted from silicon retina images and then matched, was proposed by Kogler et al. (2009) and compared to the area-based approach. The most recent work which applied classical algorithms to frames generated from event-based data is explained in Kogler et al. (2014) which featured an enhanced matching process involving belief propagation and two-stage post-filtering. Frame-based approaches have the advantage that existing algorithms can be used and they have been shown to deliver satisfactory results. However, they are strongly biased by the classical mindset of computer scientists and therefore cannot leverage the full advantages of the silicon retina. For this reason, frame-based approaches do not explicitly incorporate key principle (3) but many area-based approaches tend to be grounded on principle (1). This is because common area-based comparison metrics (such as SAD, SSD, NCC etc.) aggregate the matching cost of pixels with equal disparity. The same authors proposed an event-based algorithm (Kogler et al., 2011) that made use of the explicit timing of the events (3a) while also considering the history of past events, thus accumulating evidence over time (3b). A mechanism was implemented that selected only the best matches (2). This time-based approach has also been combined with an area-based approach to improve robustness and was implemented in real-time on a FPGA (Sulzbachner et al., 2011). A further improvement on the time-based approach of Kogler et al. (2011) was achieved by implementing a spatial aggregation mechanism of disparity evidence (1) (Eibensteiner et al., 2012). Other simple, time-based approaches – termed in this paper as "event-based approaches" – use epipolar constraints as the main matching criterion (Rogister et al., 2012; Dominguez-Morales et al., 2013). Note that by definition, all true event-based algorithms implement principle (3a). In Rogister et al. (2012) matching performance was further improved by adding additional restrictions on uniqueness, ordering, and other constraints (2). Another interesting work improved stereo matching based on epipolar constraints by using up to six cameras simultaneously (Carneiro et al., 2013). Many of these event-based approaches share the common drawback that events are sparse and non-informative. For this reason, other research has focused on generating event-based features such as orientations in order to use them as an additional matching criterion (Serrano-Gotarredona et al., 2013; Camunas-Mesa et al., 2014). Finally, very recent work (Piatkowska et al., 2014, 2013; Firouzi and Conradt, 2015), has revisited the cooperative algorithm from Marr and Poggio (1976) and applied it to data from event-based cameras. As this is closely related to the process described in Chapter 5, it will be discussed in further detail there.

4.2.2 The concept of time surfaces

The stream of events from the silicon retina can be mathematically defined as follows (Benosman et al., 2014): let $e = (\mathbf{p}, s, t)$ be a triplet given the position $\mathbf{p} = (x, y)^\top$, the time t and the polarity $s \in \{-1, 1\}$ of an event. Then, the function Σ_e maps the time t to $\mathbf{p}$, whereas the function S_e maps the polarity s to $\mathbf{p}$:

$$\begin{array}{rcl} \Sigma_e : \mathbb{R}^2 & \to & \mathbb{R} \\ \mathbf{p} & \mapsto & t = \Sigma_e \end{array} \quad , \quad \begin{array}{rcl} S_e : \mathbb{R}^2 & \to & \{-1, 1\} \\ \mathbf{p} & \mapsto & s = S_e \end{array} \tag{4.4}$$

As time is an increasing function, Σ_e is a monotonically increasing surface and corresponds to the mathematical description of a *temporal image* introduced in the previous section. Thus, a *time surface* is referred to as any arbitrary function $\Omega(\mathbf{p}, t)$ of Σ_e and S_e. In this paper, the main focus is on two examples of time surfaces, the linear-time surface (LTS) and exponential-time surface (ETS), denoted as Λ and Γ respectively. The definition for the LTS is

$$\begin{array}{rcl} \Lambda : \mathbb{R}^2 \times \mathbb{R} & \to & \mathbb{R} \\ (\mathbf{p}, t) & \mapsto & \begin{cases} S_e(\mathbf{p}) \cdot \left(1 + \frac{\Sigma_e(\mathbf{p}) - t}{\tau}\right)^+ & , \quad t \geq \Sigma_e(\mathbf{p}) \\ 0 & , \quad t < \Sigma_e(\mathbf{p}) \end{cases} \end{array} \tag{4.5}$$

where $^+$ denotes the positive part of the expression in parenthesis. Equivalently, the ETS is defined as

$$\begin{array}{rcl} \Gamma : \mathbb{R}^2 \times \mathbb{R} & \to & \mathbb{R} \\ (\mathbf{p}, t) & \mapsto & \begin{cases} S_e(\mathbf{p}) \cdot e^{\frac{\Sigma_e(\mathbf{p}) - t}{\tau}} & , \quad t \geq \Sigma_e(\mathbf{p}) \\ 0 & , \quad t < \Sigma_e(\mathbf{p}) \end{cases} \end{array} \tag{4.6}$$

In classical cameras, images are acquired by integrating the photo current in each pixel over a short time period, known as the exposure time. Similarly, events from the silicon retina could be integrated to generate images that could be subsequently processed with conventional algorithms in a classical frame-based manner. This approach was used by the first event-based machine vision algorithms and it can be mathematically expressed as a special case of time surface, namely, the static-time surface (STS)

$$\begin{array}{rcl} I : \mathbb{R}^2 \times \mathbb{R} & \to & \mathbb{R} \\ (\mathbf{p}, t) & \mapsto & \begin{cases} S_e(\mathbf{p}) & , \quad \Sigma_e(\mathbf{p}) - \frac{\Delta t}{2} \leq t \geq \Sigma_e(\mathbf{p}) + \frac{\Delta t}{2} \\ 0 & , \quad \text{else} \end{cases} \end{array} \tag{4.7}$$

where Δt denotes the time interval over which events are accumulated. The visualization of the STS is classed as a *static image* whereas visualizations of the LTS and ETS, for example, are

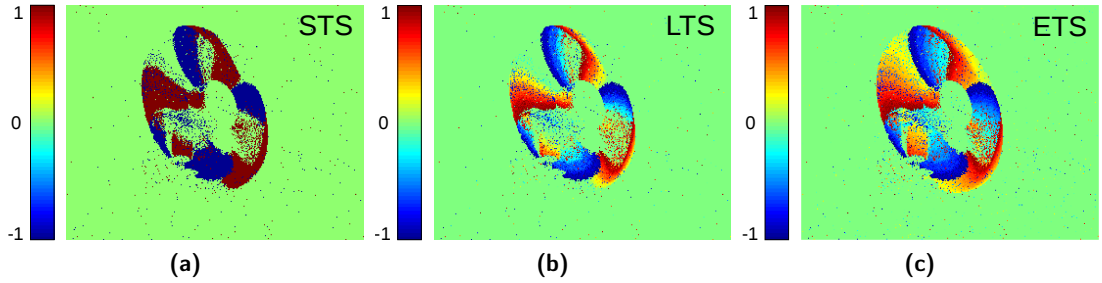


Figure 4.3: Visualization of time surfaces from a spinning fan. **(a)** Static image of the static time surface (STS) with an accumulation time of $\Delta t = 30$ ms. Visual flow cannot be inferred from this image and the effect of motion blur is visible. **(b)** Dynamic image of the linear time surface (LTS) with a time decay of $\tau = 30$ ms. **(c)** Dynamic image of the exponential time surface (ETS) with a time decay of ø = 30ms. For both of the dynamic images, the visual flow can be derived, i.e. the fan is observed to spin counter-clockwise.

termed as *dynamic images*[2]. Illustrations of the different images are shown for the example of a spinning fan in Fig. 4.3. The main advantage of dynamic images is that they contain temporal dynamics that are not present in static images. For example, the visual flow can be derived from the LTS or the ETS, but not from the STS. This is apparent when looking at the visualizations in Fig. 4.3. Another shortcoming of static images is the loss of precise spatial information caused by the accumulation process, an effect known as motion blur. Time surfaces (beyond trivial LTS) are worthy of investigation because they make it possible to weight events according to their time of occurrence. From a theoretical analysis, it can be observed that the ETS encodes the visual flow in such a way that the time surface is less variant to the magnitude of velocity compared to the LTS (see Appendix A.3.3). Thus, it may benefit the stereo matching process.

4.2.3 Spatiotemporal features

A *spatiotemporal feature* can be defined as the $w \times w$ local vicinity of Ω around the spatial position $\mathbf{p} = (x, y)^T$ at time t.

$$F_e(\mathbf{q}) = \left\{ \Omega(\mathbf{p} + \mathbf{q}, t) \mid \|\mathbf{q}\|_\infty \leq \frac{w}{2} \right\} \tag{4.8}$$

As it is fixed in time, a spatiotemporal feature is only a function of relative spatial position $\mathbf{q} = (u, v)^T$. An illustration is shown in Fig. 4.4. Note that the position $\mathbf{p}$ at which the feature is centered and the time t at which it is recorded, together, represent an event e. Thus, such a feature is always associated with an event and the notation F_e is used to denote this fact. In

[2] not to be confused with temporal images which represent the visualization of Σ_e

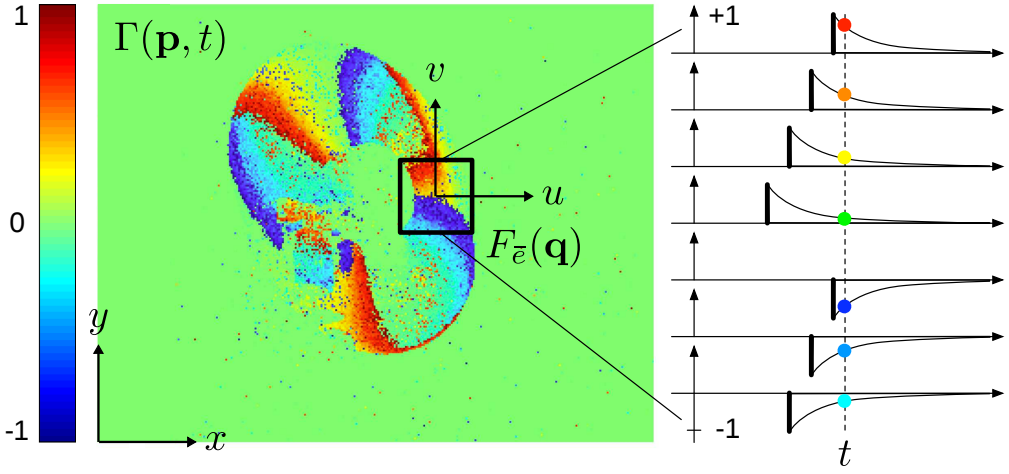


Figure 4.4: Illustration of a spatiotemporal feature taken from a time surface of a spinning fan. In this example, the time surface is an ETS. The spatiotemporal feature is outlined by the black frame. An inset on the right illustrates the evolution of the time surface of some random points along the vertical axis in the feature's vicinity. The process of feature extraction is visually depicted as local sampling of the time surface at time t.

addition, the mean μ_F of a spatiotemporal feature can be defined as

$$\mu_F = \frac{1}{w^2} \sum_{\mathbf{q}} F_e(\mathbf{q}) \tag{4.9}$$

and accordingly the standard deviation σ_F as

$$\sigma_F = \sqrt{\sum_{\mathbf{q}} \left(F_e(\mathbf{q}) - \mu_F\right)^2} \tag{4.10}$$

4.2.4 Coarse temporal matching

As suggested by the early event-based algorithms, the main advantage of event-based stereo vision is that events from two spatially separate retinas can be coarsely matched by their time of occurrence. This reduces the correspondence problem significantly in comparison to the classical frame-based approach, whereby all of the pixels in both frames need to be matched simultaneously. Simple algorithms that chiefly rely on this matching criterion work fine as long as there is only little activity in the scene. This would be the case when there is a single moving object, for example. Conversely, multiple moving objects or movement of the sensors generate many simultaneous events resulting in ambiguity during the matching process and an inaccurate 3D reconstruction as a result.

4.2.5 Fine spatiotemporal matching

To address the fact that the coarse temporal matching criterion only works well in very simple scenes, an additional step can be introduced into the matching process. Once two events from different sources have been matched according to the coarse temporal matching criterion, a measure of the degree of correlation between associated spatiotemporal features can be used as a further matching criterion. This study adopts the normalized cross-correlation (NCC) metric, but similar measures can be used interchangeably:

$$\rho(e_L, e_R) = \frac{\sum_{\mathbf{q}} \left(F_{e_L}(\mathbf{q}) - \mu_{F_L}\right)\left(F_{e_R}(\mathbf{q}) - \mu_{F_R}\right)}{\sigma_{F_L}\sigma_{F_R}} \tag{4.11}$$

where μ_{F_L} and μ_{F_R} are the mean values of the associated spatiotemporal features and σ_{F_L} and σ_{F_R} are their standard deviations.

4.2.6 Event-based STC stereo algorithm

Thus far, the prerequisites for the development of an event-based stereo algorithm have been elaborated. As one of the central techniques involves correlation-based matching of spatiotemporal features, the algorithm will be referred to as "the spatiotemporal correlation (STC) stereo algorithm". A concise explanation of the algorithm can be found in the following paragraph, while a brief mathematical formulation is provided below.

In a first step, each event from either source is added to its corresponding temporal image Σ_{e_L} and Σ_{e_R} respectively. From this point on, only events from the left source are further processed. For each such event e_L^i, the set $S_R(e_L^i)$ of possible matching events e_R^j from the right source is considered to contain all events of the same polarity $s_i = s_j$, which occurred within a fixed time window t_w, i.e. $t^i - t^j \leq t_w$. The epipolar constraint is then enforced by choosing a subset $M_R(e_L^i) \subset S_R(e_L^i)$, whereby events within a minimum distance to the epipolar line $\mathbf{l}^i = F \cdot \begin{pmatrix}\mathbf{p}^i \\ 1\end{pmatrix}$ are selectively included, where F is the fundamental matrix. The spatiotemporal features of event e_L^i and all events e_R^j from $M_R(e_L^i)$ are extracted and their NCC is computed, as described in equation 4.11. The pair of events with the highest NCC is considered a match, while all others are deemed to be false targets and consequently discarded. The final output is obtained by triangulating this pair of events, each of which produces a 3D event $e_{3D} = (\mathbf{P}, s, t)$ with a spatial 3D position $\mathbf{P} = (X, Y, Z)^\top$, polarity s and time of occurrence t.

4.3 Evaluation methods for event-based stereo vision algorithms

In order to apply epipolar constraints, the extrinsic parameters (fundamental matrix) of the stereo setup need to be known. This is one reason why cameras need to be calibrated prior to applying the algorithm detailed in this study. Another reason is that full calibration (intrinsic

Algorithm 1 Event-based STC stereo algorithm

Require: Stereo setup with event-based cameras C_L, C_R
Require: Estimation of the fundamental matrix F

for all events e_L^i from C_L and e_R^j from C_R **do**
  $\Sigma_{e_L}(\mathbf{p}^i) = t^i$ ▷ *Update left temporal image*
  $\Sigma_{e_R}(\mathbf{p}^j) = t^j$ ▷ *Update right temporal image*
  if $e_{L,R}^{i,j}$ is from retina C_L **then**
    $S_R(e_L^i) = \{e_R^j \mid |t^i - t^j| \leq t_w\}$ ▷ *Set of possible matches*
    $\mathbf{l}^i = F \cdot \begin{pmatrix}\mathbf{p}^i \\ 1\end{pmatrix}$ ▷ *Epipolar line*
    Find $M_R(e_L^i) \subset S_R(e_L^i)$ ▷ *Epipolar constraint*
    Compute $F_{e_L^i}$ from $\Omega_L(\mathbf{p}^i, t^i)$ ▷ *Left spatiotemporal feature*
    for all events $e_R^j \in M_R(e_L^i)$ **do**
      Compute $F_{e_R^j}$ from $\Omega_R(\mathbf{p}^j, t^j)$ ▷ *Right spatiotemporal feature*
      Compute $\rho(e_L^i, e_R^j)$ ▷ *Cross-correlation*
    end for
    Add best match $m = (p^i, p^j, t^i)$ to the list of found matches $T(m^n)$
  end if
end for
return $T(m^n)$

and extrinsic parameters) is required for 3D reconstruction. This is particularly necessary in order to compare the output of the algorithm with ground-truth data represented by absolute world coordinates. Little of the existing literature details how to calibrate event-based cameras (Benosman et al., 2011; Camunas-Mesa et al., 2014; Censi and Scaramuzza, 2014; Mueggler et al., 2014) and expensive setups (e.g. array of blinking LEDs) or monitors are often used. Normal frame-based stereo correspondence algorithms are usually evaluated with the Middlebury stereo datasets[3]. In this case, the percentage of "bad pixels" with an error greater than a given threshold (typically 0.5, 1.0, 2.0 or 4.0) is used as a metric for the performance of stereo matching (Scharstein and Szeliski, 2002). These datasets (Scharstein and Szeliski, 2003; Scharstein and Pal, 2007; Hirschmuller and Scharstein, 2007) contain pixel accurate ground-truth disparity maps that were acquired with a structured lighting technique. Unfortunately, there are no equivalent datasets and conventional metrics available for the evaluation of event-based stereo algorithms. Much of the relevant research compares algorithms or relies solely on qualitative evaluation of depth or disparity maps. One exception is a very recent work that uses an event-based stereo vision system together with a conventional video-camera stereo vision system to produce reference data (Kogler et al., 2013).

[3] http://vision.middlebury.edu/stereo/data/

4.3.1 Sensor calibration and triangulation of stereo events

Throughout this research, different event-based stereo systems are used which were calibrated according to different techniques. The new generation of event-based cameras also provide grayscale images, so that they can be calibrated like normal cameras. This is not the case for purely event-based cameras, which require a different approach. Thus, a distinction must be made between frame-based and event-based calibration methods. To address this, two event-based calibration methods have been developed which are briefly summarized here and explained in more detail in Appendix A.2. In the first method, calibration images are generated from histograms of accumulated events captured by event-based cameras mounted on a vibrating substrate. The advantage of this method is that it is compatible with normal cheap calibration models (e.g. a checkerboard pattern printed on a paper). Unsurprisingly, this method does not yield very accurate results, but they are certainly sufficient, as evidenced by the typical mean reprojection errors (e_{RME}) of 0.4 pixels. The second method is based on calibration images, which stem from an accumulation of events generated from a blinking calibration pattern on an LCD monitor, which yielded more accurate results ($e_{RME} \approx 0.25$). However, a frame-based approach using two DAVIS240b sensors provided the most accurate results ($e_{RME} \approx 0.15$). Fig. 4.5 shows examples of the calibration images used and the calibration results for each method.

To obtain 3D events in absolute world coordinates, matched pairs of events are triangulated. Theoretically, triangulation could be achieved by determining the intersection of back-projected rays from the spatial positions of the events. In practice, however, these rays will generally not intersect due to noise in the spatial positions of the matched events. Thus, it is necessary to estimate this ideal scenario, which can be achieved using many different methods (Hartley and Zisserman, 2004). Here, a simple linear triangulation method is employed, which is described in Appendix A.1.2.

4.3.2 Model-based ground-truth evaluation

The model-based ground-truth evaluation method adopted here is similar to that described in Carneiro et al. (2013). Firstly, matched events are triangulated from both cameras to produce 3D events. Afterwards, the 3D events are binned into short time slices, typically 30 ms, each of which forms a reconstructed 3D point cloud of the scene at a particular point in time. Each of these point clouds is then sequentially registered with a 3D model of the scene which is presumed to be perfectly accurate. In order to register the point cloud with the 3D model, an efficient ICP algorithm is employed which uses a k-dimensional tree search to complete the task (Rusinkiewicz and Levoy, 2001). The ground truth at each point in time is assumed to correspond to the optimal pose of the 3D model, whereby the Euclidean distance between the point cloud and the model is minimized. To evaluate the performance of the stereo matching algorithm, the percentage of correct matches (PCM) is used as a metric. A 3D event is considered to be a correct match when its distance to the closest point of the 3D

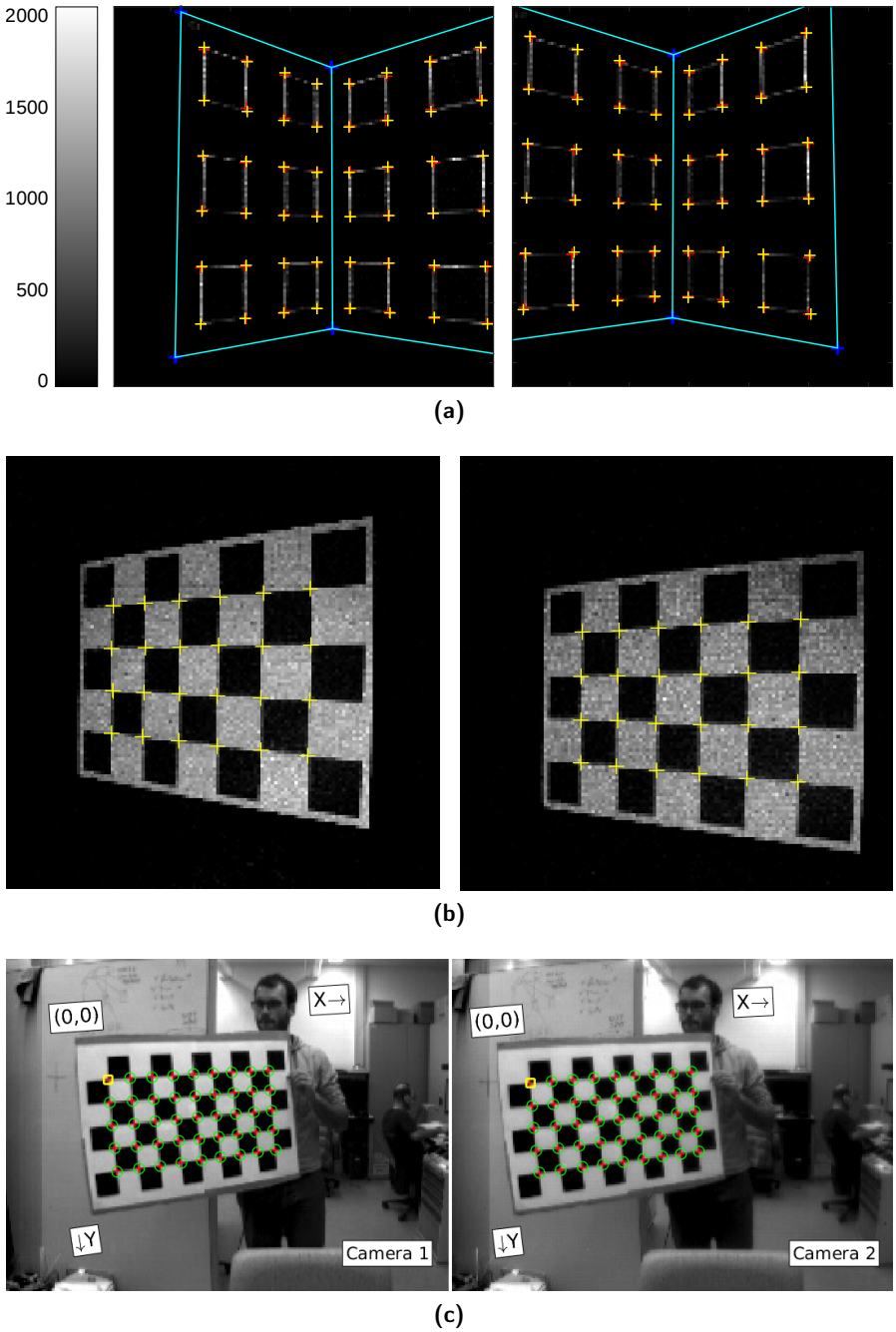


Figure 4.5: Methods for stereo calibration of event-based cameras. **(a)** Vibrating sensors method (see Appendix A.2.1). The images generated are event histograms accumulated during $T = 2$ s. The corners of the calibration model, which are manually detected, are indicated in red and their reprojections in yellow. **(b)** Flashing patterns method (see Appendix A.2.1). Calibration images generated by accumulating the event stream of a flashing pattern. Yellow crosses indicate corners which are automatically detected with OpenCV. **(c)** Frame-based calibration method (see Appendix A.2.2). Screenshot of Matlab's Stereo Camera Calibrator App showing checkerboard points which are automatically detected (green circles) together with their reprojections (red crosses) in corresponding grayscale images taken with two DAVIS240b sensors.

model is smaller than the depth error in stereo ϵ_z. This depth error can be derived from the depth-disparity relation $Z = \frac{bf}{d}$, where b is the baseline and f the focal length (Gallup et al., 2008):

$$\epsilon_Z = \frac{bf}{d} - \frac{bf}{d+\epsilon_d} = \frac{Z^2\epsilon_d}{bf + z\epsilon_d} \approx \frac{Z^2}{bf}\epsilon_d \tag{4.12}$$

Here, $\epsilon_d = 1$ is set corresponding to the minimal matching error in pixels. The PCM metric is then defined as

$$PCM = \frac{1}{N_t}\sum_{i=1}^{N_t} h(e_{3D}^i) \quad \text{with} \quad h(e_{3D}) = \begin{cases} 1, & \min\limits_j(d_j(\mathbf{P},\mathbf{X}_j)) \leq \epsilon_Z \\ 0, & \min\limits_j(d_j(\mathbf{P},\mathbf{X}_j)) > \epsilon_Z \end{cases} \tag{4.13}$$

where $d_j(\mathbf{P},\mathbf{X}_j)$ is the Euclidean distance between the spatial position of a 3D event $\mathbf{P}$ and the j-th point of the 3D model $\mathbf{X}_j$ and N_t, the total number of binned 3D events in a time slice t .

4.3.3 Measurement-based ground-truth evaluation

When evaluating complex dynamic scenes, it is hard to model the ground truth, but it can be measured. One possibility would be to use a reference video-camera stereo system that provides accurate disparity or depth maps. Such a method was recently introduced by Kogler et al. (2013). Another possibility is to use a more reliable 3D acquisition technique such as structured light. Here, a Microsoft Kinect is used to acquire 3D ground-truth data. In order to register 3D events with the ground truth which has been measured, the relative position of the Kinect with respect to the event-based camera stereo system needs to be precisely known. This can be obtained from the stereo calibration of the Kinect sensor and one of the event-based cameras. Again, matched events are triangulated and binned into 30 ms time slices. In so doing, the reconstructed 3D point clouds are sequentially compared to the ground-truth point clouds, which have been acquired and temporally synchronized (obtained from one frame of the Kinect running at 30 Hz). The 3D error $\epsilon_{\mathbf{X}}$ for each 3D event, representing a point in the reconstructed point cloud, is computed as the minimum distance between any 3D point and the temporally coinciding ground truth.

Disparity error and matching performance metric

To assess the matching performance of the algorithm advocated in this paper, the error in disparity ϵ_d is calculated. This is a more meaningful metric than the 3D error $\epsilon_{\mathbf{X}}$ because it does not vary for objects located at different depths. To determine the disparity error, the error in depth ϵ_Z needs to be measured. It is assumed that $\epsilon_Z \approx \epsilon_{\mathbf{X}}$ which is generally not correct. Theoretically, ϵ_Z should be measured as the distance from a reconstructed 3D point to its

nearest neighbor in the ground-truth data, along a line of sight determined by the position of the 3D point and the center of the camera. This is the only way to ensure that the depth error is computed from physically corresponding points, rather than nearest neighbors which do not represent the same point. However, due to the poor resolution of the DAVIS sensor (which results in poor depth resolution) and an additional error introduced by the fact that the Kinect frames and 3D events are not perfectly temporarily synchronized, qualitative evaluations show that $\epsilon_X \approx \epsilon_Y \ll \epsilon_Z$ and therefore the assumption $\epsilon_Z \approx \epsilon_{\mathbf{X}}$ holds. From Eq. 4.12, the disparity error can be computed from the depth error as follows:

$$\epsilon_d = \frac{bf}{Z^2}\epsilon_Z \tag{4.14}$$

As previously discussed, the metric to evaluate the performance of stereo matching is based on the PCM, which can now be defined as:

$$PCM = \frac{1}{N_t}\sum_{i=1}^{N_t} h(e_{3D}^i) \quad \text{with} \quad h(e_{3D}) = \begin{cases} 1, & \epsilon_d \leq 1 \\ 0, & \epsilon_d > 1 \end{cases} \tag{4.15}$$

4.4 Experiments and results

4.4.1 Parameter analysis

In the first experiment, the effect of the main parameters on performance, robustness and precision is studied. These parameters include window size ω, decay constant τ and type of time surface Ω. As a result, the scene must be neither too simple, nor too complex, in order to be suitable. If the scene is too simple, as would be the case for a moving bar for example, the stereo matching is trivial. Conversely, a complex scene makes it too difficult to validate performance and is more likely to lead to artefacts that are specific to the scene. A trade-off was found, which involved a scene consisting of a wire-framed cube moving towards the cameras while spinning around an arbitrary axis near its center of gravity. Since the cube is "transparent", arbitrary disparity gradients can be generated ranging from zero (parallel to baseline) to infinity (perpendicular to baseline). The spinning causes the edges of the cube to move in all kinds of directions at various velocities, which produces a highly variable visual flow. Generally, it would be expected that any stereo matching method based on correlation will perform worse with regard to disparity gradients exceeding the disparity-gradient limit, as the topology of the scene is reversed in each view (Trivedi and Lloyd, 1985; Filippini and Banks, 2009). The matching algorithm developed in this study, however, is based on correlating not only spatial information, but also visual flow. The latter is strongly present in this dynamic scene and might help to overcome the disparity gradient limit. For this experiment, a model-based ground-truth evaluation method was used. The ground-truth data was ascertained by finding the best shape registration of the model and the data at any given point in time, as

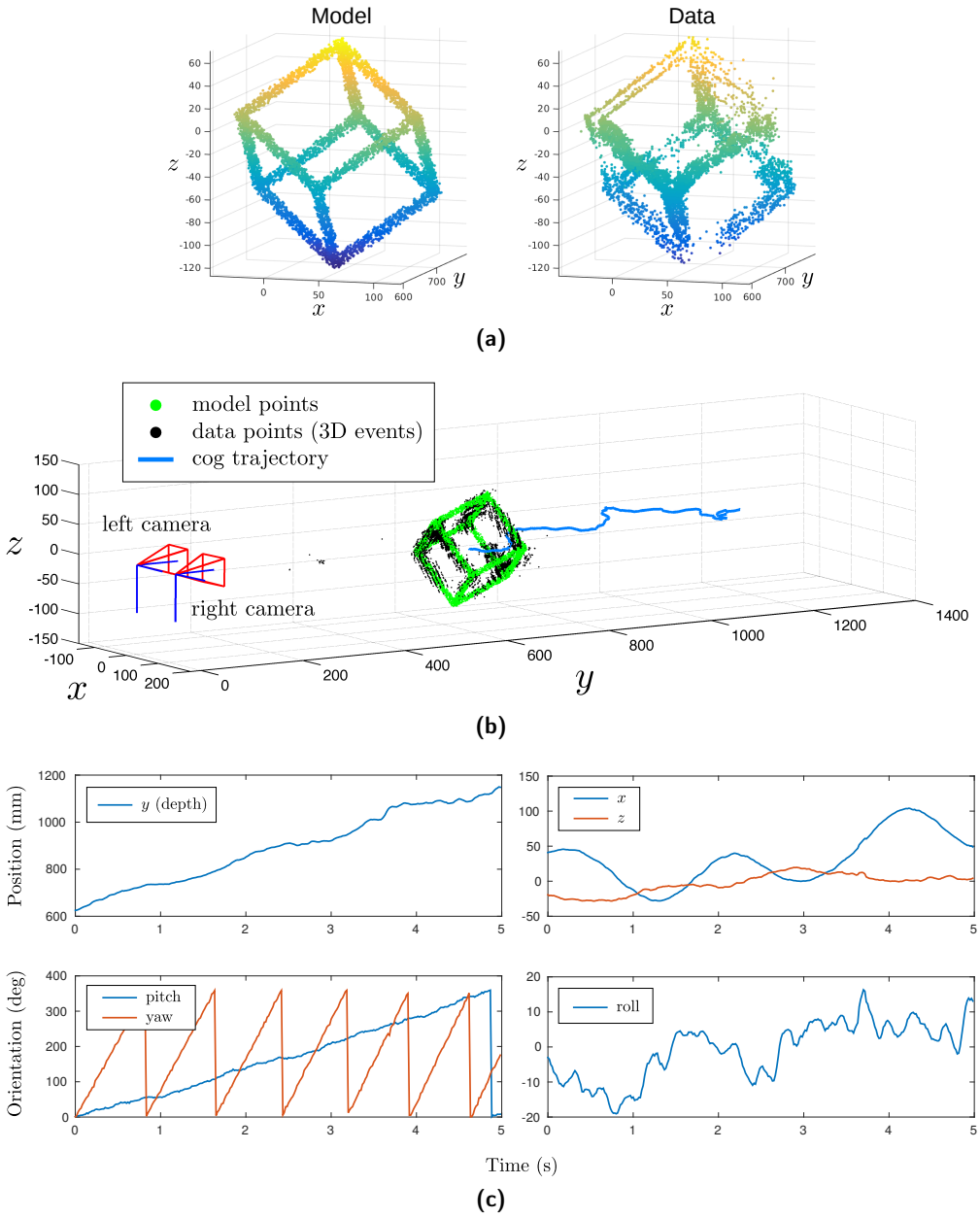


Figure 4.6: Model-based ground-truth evaluation of the stereo matching algorithm. **(a)** 3D model of a wire-framed cube with side length $l = 12$ cm next to its 3D reconstruction. **(b)** Visualization of the dynamic scene. The reconstructed trajectory (3D position) of the model nicely shows how it moves through space. The initial pose of the model at $t = 0$ s is rendered in green and its corresponding reconstruction in black. Here, time bins of 16.7 ms are used to register the shape of the model and data. **(c)** 6-DOF pose estimation of the model plotted as trajectories of position and orientation for the full duration of the experiment ($T = 5$ s). As can be observed, the model's axis of rotation is mainly in the direction of z (yaw) with some minor y component (pitch). Time resolution is 16.7 ms.

previously described in Section 4.3.2. The outcome is shown in Fig. 4.6c, which illustrates the continuous 6-DOF pose (position and orientation) of the model over time.

Coarse temporal matching window

In order to conduct coarse temporal matching successfully, it is crucial to choose a time window of an appropriate length. The window has to be long enough to ensure that the majority of corresponding events from both sensors are captured. Conversely, it should be as short as possible to minimize ambiguity. The accuracy of the timing between sensors depends on environmental factors such as the lighting conditions, viewing angle, and the contrast of the stimuli, but also intrinsic artifacts such as the sensor's bias settings, or the arbitration and communication of events. Although intrinsic artifacts can cause temporal jitter, this usually remains below a few microseconds (Lichtsteiner et al., 2008; Posch et al., 2010a; Berner et al., 2013). Conversely, environmental artifacts can cause temporal noise which lasts milliseconds. For example, the disparate viewing point of the cameras might mean that the same physical object has different levels of contrast. This causes unequal photo currents at the corresponding pixels and the timing of events falls out of sync as a result. Note that it is even possible that corresponding events could have diverse polarity. However, it can be quantitatively confirmed that this is only the case for a minority of events, in particular, if the baseline is not very large. For the majority of events the polarity constraint (Rogister et al., 2012) is valid. The experiments reveal that the time window should be in the range of a few milliseconds and it is largely unaffected by the content of the scene, except for very slowly moving objects. This case, however, is only of limited interest given that event-based cameras are more likely to be used in rapidly changing scenarios.

Spatiotemporal feature size

Spatiotemporal features are characterized by the decay constant τ and the size of their neighborhood which is defined as a quadratic window of side length ω centered at the events spatial position (see Eq. 4.8). The optimal window size for models of stereo correspondence based on cross-correlation has been studied extensively in the existing literature, which suggests that it depends on the density of the samples in the stimuli (Banks et al., 2005). If the window is too small, the disparity estimate is poor because of the ambiguity of small features. On the other hand, the window should also not be too large because if the disparity changes significantly within the window, the cross-correlation will yield the average disparity and fail to detect variation in disparity. The idea of using various window sizes in a hierarchical manner (coarse to fine), the so-called multi-resolution approach, has been widely adopted and has been shown to mitigate the matching problem. However, for the purposes of this study, fixed window sizes proved to be satisfactory.

Fig. 4.7 shows the effect of varying the size of the spatial window. The reconstructed data has been evaluated every 16.7 ms and the PCM as well as the mean reconstruction error MRE are

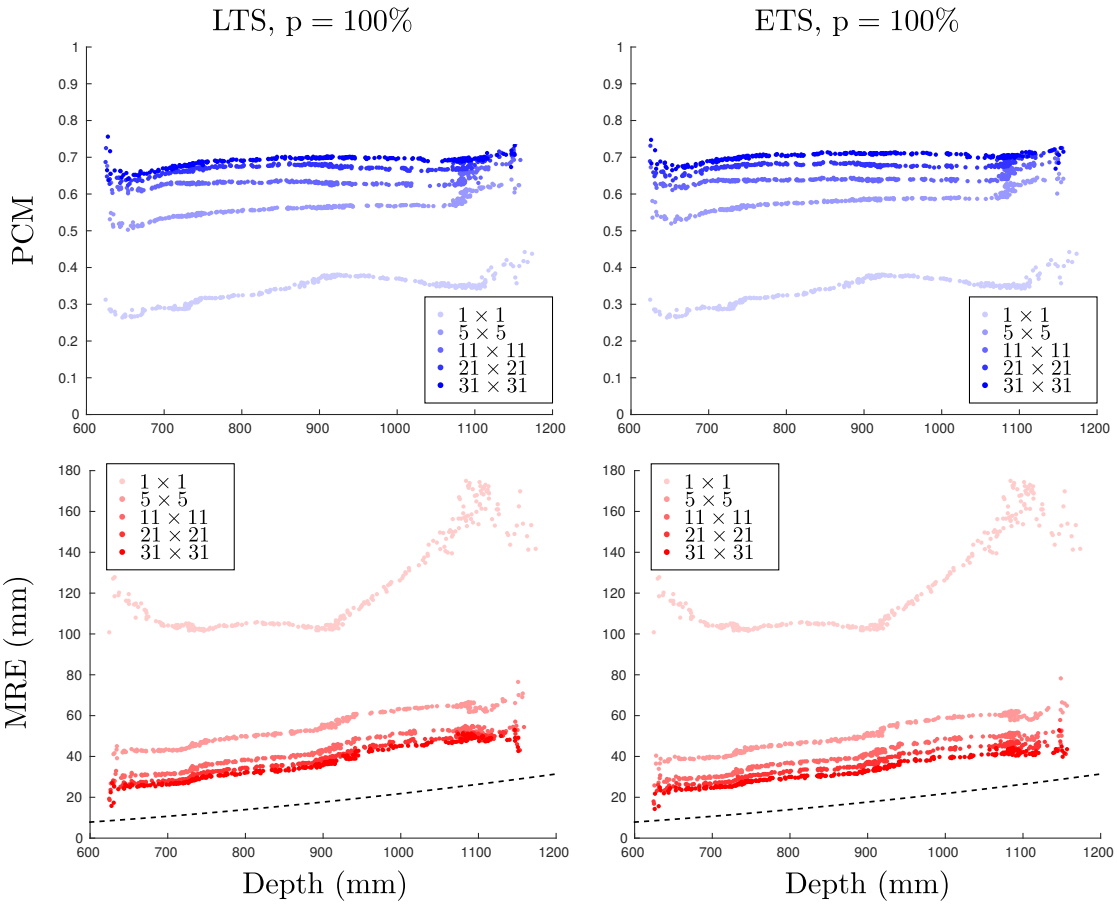


Figure 4.7: Performance of the stereo matching algorithm with varying feature size. The PCM is shown at the top (blue) while the bottom plots depict the mean reconstruction error MRE (red). The dashed line denotes the depth error $\epsilon_y = y^2/bf$ corresponding to the minimum matching error of one pixel. The acceptance rate $p = 100\%$ indicates that no 3D events have been rejected. The results for the LTS and ETS are shown on the left and right respectively. Both time surfaces have a fixed time decay of $\tau = 1e^{-4}$.

plotted against the model's position in depth. The results have been smoothed in order to allow for better comparison among them. As expected, this study confirms that performance increases as the size of the spatial window grows. The case where the window consists of a single pixel, is equivalent to the implementation in Rogister et al. (2012) whereby information from the local neighborhood is not considered. Even a small spatial window of 5×5 pixels already results in significant improvement in the PCM and further extension brings it up to 70%. The PCM is fairly robust, covering the entire range of depth from 600 mm to 1200 mm. The increasing MRE is explained by the simple geometry of stereo resolution, which decreases with depth. The depth error $\epsilon_y = y^2/bf$, which corresponds to the minimum matching error of one pixel, is indicated by the dashed line. Recall that this value is used as a threshold to classify correct and false matches. Even for a PCM as high as 70%, the MRE is well above the threshold which suggests that the fewer false matches are located far from the ground truth.

Interestingly, both the PCM and MRE continuously improve even in the case of very large window sizes such as 31×31 pixels, which covers nearly 50% of the entire cube for $y > 1100$ mm. This, together with the fact that the PCM stays nearly constant while the projected size of the cube changes significantly, suggests that the approach adopted in this study is not as sensitive to projective distortions arising from larger windows as might be expected from the experience of other matching algorithms based on correlation (Kanade and Okutomi, 1994). This could be ascribed to the additional visual information present in the spatiotemporal features. Different types of time surfaces (linear versus exponential) do not seem to have any effect when the window size is varied.

Spatiotemporal feature decay constant

A general problem of time surfaces are the dips generated by events that happened far in the past. Such outliers have a significant influence on the local variance. As a result, it may change the correlation coefficient such that an area which was initially correlated becomes completely uncorrelated. It could be argued that these outliers are rare because they are likely to be overwritten by newer activity. However, a similar problem tends to exist for regions that contain information from the distant past. Consider a local area in a temporal image, with an edge, caused by activities which occurred at different points in time. This temporal edge will always yield a high correlation with others of its kind, almost regardless of the spatial and temporal structure on both of its sides. In this case, relevant spatiotemporal information might be crowded out by a less informative coincidence of temporarily separate activities. In general, it can be stated that the correlation coefficient is influenced most by the highest temporal difference, whereas smaller differences are relatively less significant. This explains the decision to introduce time surfaces, which deal with this problem by omitting the very distant history (LTS) or compressing it so that it becomes less influential (ETS).

The decay constant τ, introduced by the LTS, enables the researcher to control the influence of older activity. Here, τ defines the maximum possible age of an event. The optimal choice for τ depends on the spatial density of events and the visual flow. From a theoretical point of view, it should be chosen such that the space between spatial edges is filled with temporal information — i.e. the motion gradient should fully decay between the edge where it originates and the neighboring edges. If f is the spatial frequency of events and v their velocity, then the optimal time decay would be

$$\tau = \frac{v}{f} \tag{4.16}$$

Obviously, this is an oversimplification of a natural scenario, as the spatial frequency and the velocity of events vary strongly within a scene. Nevertheless, it shows the way τ tends to relate to the scene characteristics. Now, it should also become clear why the ETS was introduced. In contrast to the LTS, no information is discarded when an ETS is adopted. Instead, past events

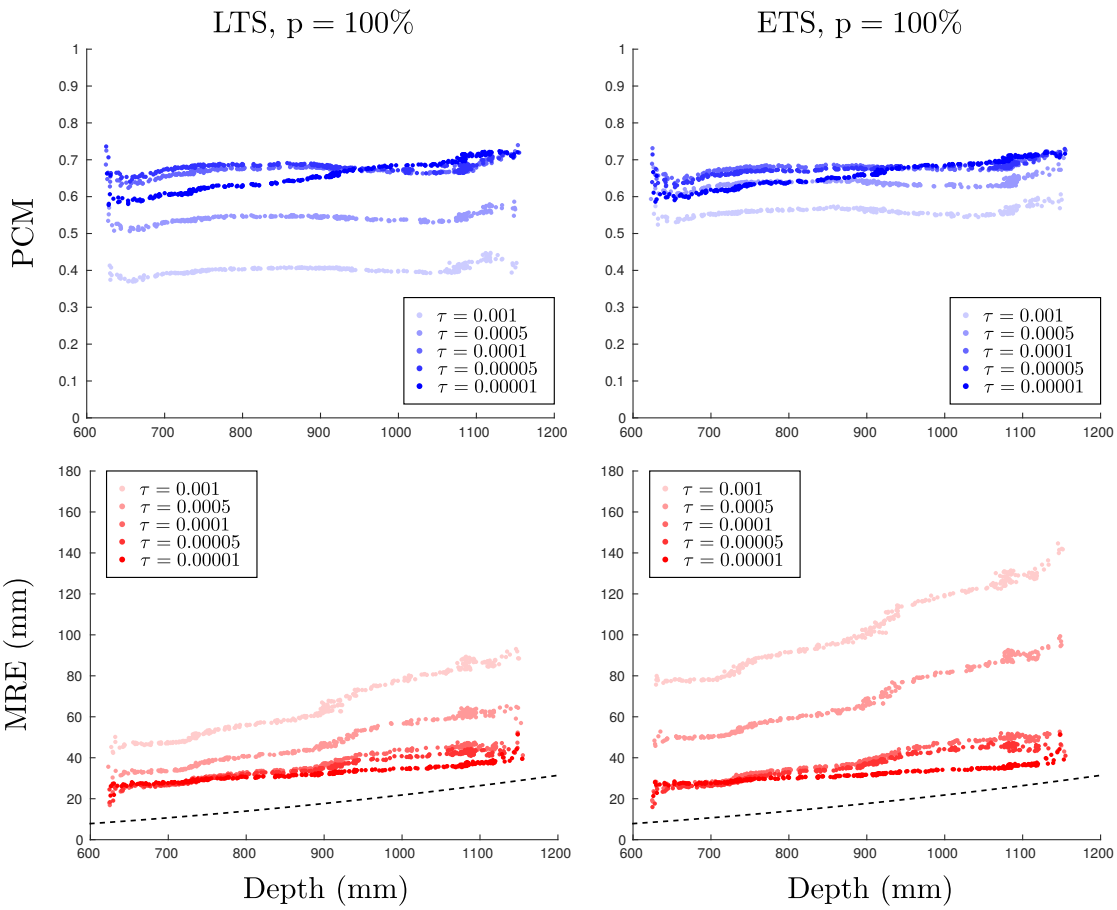


Figure 4.8: Performance of the stereo matching algorithm with varying decay constant. The PCM is shown on the top (blue) while the bottom plots depict the mean reconstruction error MRE (red). The dashed line denotes the depth error $\epsilon_y = y^2/bf$ corresponding to the minimum matching error of one pixel. The acceptance rate $p = 100\%$ indicates that no 3D events have been rejected. The results for the LTS and ETS are shown on the left and right respectively. In both cases, a fixed spatial window size of 21×21 was used.

are compressed in an exponentially decreasingly manner, meaning that they become less important over time. The optimal decay time depends on the same factors as in the previous case, but the ETS is likely to be a better choice for scenes involving a high degree of variance in visual flow and spatial density.

The results of varying the decay constant with a fixed spatial window size are shown in Fig. 4.8. At the beginning, the PCM and MRE improve as τ decreases, as expected given that a longer history of events is considered. At some point (in this case, $\tau > 0.00005$) old events start to negatively affect performance due to the effects described above. This is particularly evident when the cube is near to the cameras. This phenomenon can be explained as follows: The size of the projection of the cube grows as it approaches the cameras, meaning that events appear to move faster. As can be ascertained from Eq. 4.16, this has the same effect as raising τ. Thus, both observations are consistent with each other, describing the initial improvement of

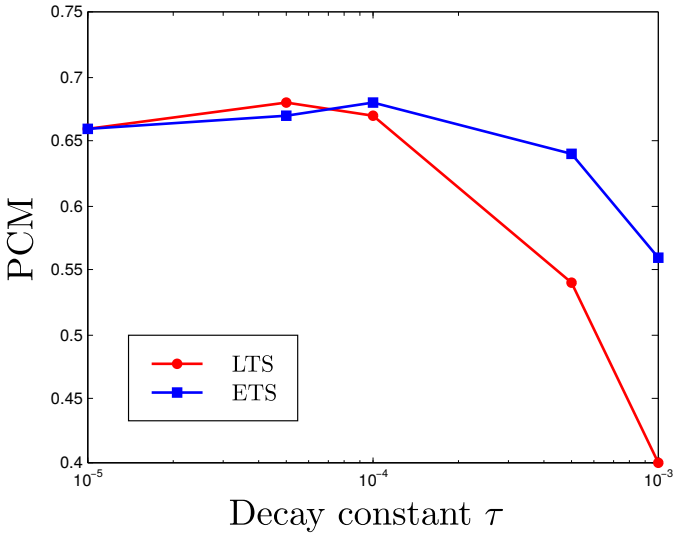


Figure 4.9: A comparison of how different types of time surfaces affect the performance of stereo matching. The average PCM for both types of time surfaces are plotted in relation to the decay constant τ. As the ETS spans a wider range, it outperforms the LTS. Each point was obtained by averaging the results of Fig. 4.8 over all points in depth

stereo matching for smaller values of τ. The effect can also be observed in the MRE, when the smallest value of $\tau = 0.00001$ is examined. At lower depths, the error is much further from the minimal depth ϵ_y (dashed line) compared to larger depths, where the MRE gets really close to ϵ_y.

When the size of the spatiotemporal feature was varied in the previous analysis, no specific characteristics could be observed for different types of time surfaces. In the present case, however, a remarkable difference can be observed when comparing the types. While the ETS performs significantly better for larger values of τ, it shows no inferiority for smaller values of τ. This effect is more clearly illustrated when the average of all positions in depth are plotted in the same graph, as illustrated in Fig. 4.9. The ETS clearly performs better over a wider range of τ, meaning that it has a higher dynamic range for velocities and thus, is the preferred choice.

Correlation coefficient threshold

No data has been rejected in the analysis thus far, meaning that every single event from one retina has been matched with an event from the other retina. The results can be easily further improved by rejecting matches with a correlation coefficient below a certain threshold. The results are listed in Tab. 4.1 and Tab. 4.2. Here, the threshold was chosen such that the best 100%, 75% and 50% (acceptance rate p) of all matches are retained respectively and the rest are discarded. It can be observed that with an acceptance rate of $p = 75\%$, the performance can be significantly improved, but that discarding further events shows only a marginal effect.

Window size $m \times m$	1×1	5×5	11×11	21×21	31×31
ETS, p = 100%	0.34	0.58	0.64	0.68	**0.70**
ETS, p = 75%	0.34	0.65	0.71	0.75	**0.78**
ETS, p = 50%	0.34	0.68	0.72	0.76	**0.80**
LTS, p = 100%	0.34	0.56	0.63	0.67	**0.69**
LTS, p = 75%	0.34	0.63	0.71	0.75	**0.78**
LTS, p = 50%	0.34	0.66	0.72	0.77	**0.81**

Table 4.1: The PCM for different acceptance rates with varying spatial window size and fixed decay constant $\tau = 1e^{-4}$.

Decay constant	0.001	0.0005	0.0001	0.00005	0.00001
ETS, p = 100%	0.56	0.64	**0.68**	0.67	0.66
ETS, p = 75%	0.62	0.72	**0.75**	0.74	0.72
ETS, p = 50%	0.65	0.74	**0.76**	0.76	0.74
LTS, p = 100%	0.40	0.54	0.67	**0.68**	0.66
LTS, p = 75%	0.42	0.60	0.75	**0.76**	0.71
LTS, p = 50%	0.44	0.64	**0.77**	0.76	0.73

Table 4.2: The PCM for different acceptance rates with varying decay constant and a fixed spatial window size of 21×21 pixels.

4.4.2 Natural scenes

The event-based STC stereo algorithm was also tested on natural scenes. A few examples of reconstructions of natural scenes are shown in Fig. 4.10 and the parameters that have been used are listed in Tab. 4.3. In the first scene (Fig. 4.10a), the hand was oriented parallel to the baseline. This generated only small disparity gradients, yielding a very good result. Conversely , the second scene (Fig. 4.10b) consists of a spinning fan that was tilted in depth in order to assess performance in scenes with larger disparity gradients. A qualitative observation can be made that the outcome is very good, which can be explained by the distinct motion cues and low number of edges present in that particular scene. Finally, a combination of plenty of ambiguity and varying disparity gradients were introduced in the last scene, featuring a moving face (Fig. 4.10c). As can be seen, the whole face is clearly reconstructed and even includes the upper part dominated by curly hair.

Ground-truth evaluation of a natural scene

The reconstructions shown so far were only qualitatively evaluated through visual inspection. For the sake of completeness, another natural scene was recorded, and a complete quantitative evaluation of the ground-truth data was performed as described in Section 4.3.3. The results

Parameters	Window size	Decay constant τ	Time surface	Acceptance rate p
Waving hand	31×31	0.00004	ETS	92%
Spinning fan	21×21	0.0001	ETS	80%
Moving face	31×31	0.00002	ETS	78%

Table 4.3: Parameters used for the reconstructed scenes in Fig. 4.10.

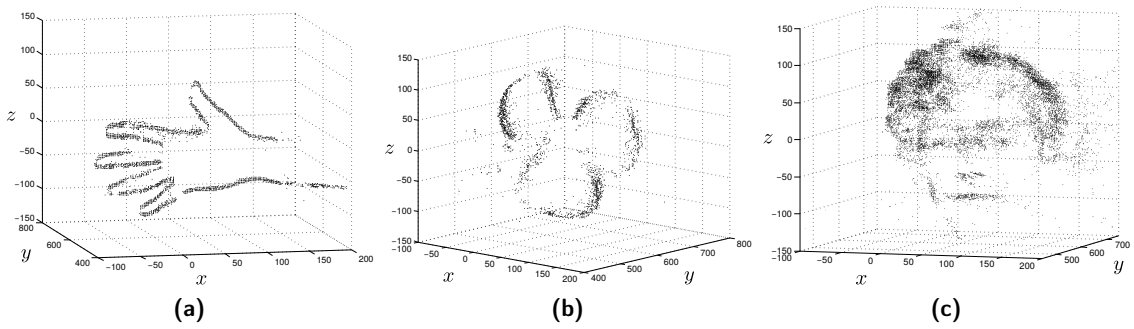


Figure 4.10: A 3D reconstruction of natural scenes. **(a)** A waving hand oriented parallel to the base-line. **(b)** A rapidly spinning fan tilted in depth. **(c)** A moving face with curly hair. The reconstructions are also provided in MATLAB files for better visualization

are summarized in Fig. 4.11. It can be observed that the stereo matching algorithm performs very well, as evidenced by the small local average disparity error of $\epsilon_d < 1$ pixels throughout the entire duration of the scene. The disparity error also remains largely constant even when the event rate of the input is highly variable, which suggests that the algorithm is fairly robust. The entire scene produced a total amount of 625,699 3D events, each of which yielded a disparity error of ϵ_d, which was computed as explained in Section 4.3.3. Fig. 4.12 shows the histogram of disparity errors. The distribution closely fits the form of a half-normal distribution

$$f(\epsilon_d) = \frac{\sqrt{2}}{\sigma\sqrt{\pi}} \exp\left(-\frac{\epsilon_d^2}{2\sigma^2}\right) \qquad \epsilon_d > 0 \tag{4.17}$$

with a mean of $\mu = \sigma\sqrt{2}/\sqrt{\pi}$, suggesting that the majority of false matches ($\epsilon_d > 1$) are only slightly displaced from their true target. Using the performance metric proposed for this evaluation method, the algorithm reached a remarkable PCM of 84.6%. Given the distribution shown in Fig. 4.12, it can be argued that completely false matches (whereby similar but physically non-corresponding features are matched) are events with $\epsilon_d > 3$ pixels. The fraction of such occurrences is only 1%.

4.5 Discussion

In this chapter, the implications of asynchronous sampling strategies on the stereo correspondence problem were considered. It was concluded that collecting spatiotemporal information proves beneficial in this regard. Time surfaces are a useful mathematical concept to represent space-time information and their analogy to normal images allows them to be used with conventional algorithms (even though this might not be efficient). When applied to time surfaces, simple image operations can compute very different but useful properties. As an example, the spatial derivative of the LTS is inversely proportional to the visual flow (Benos-

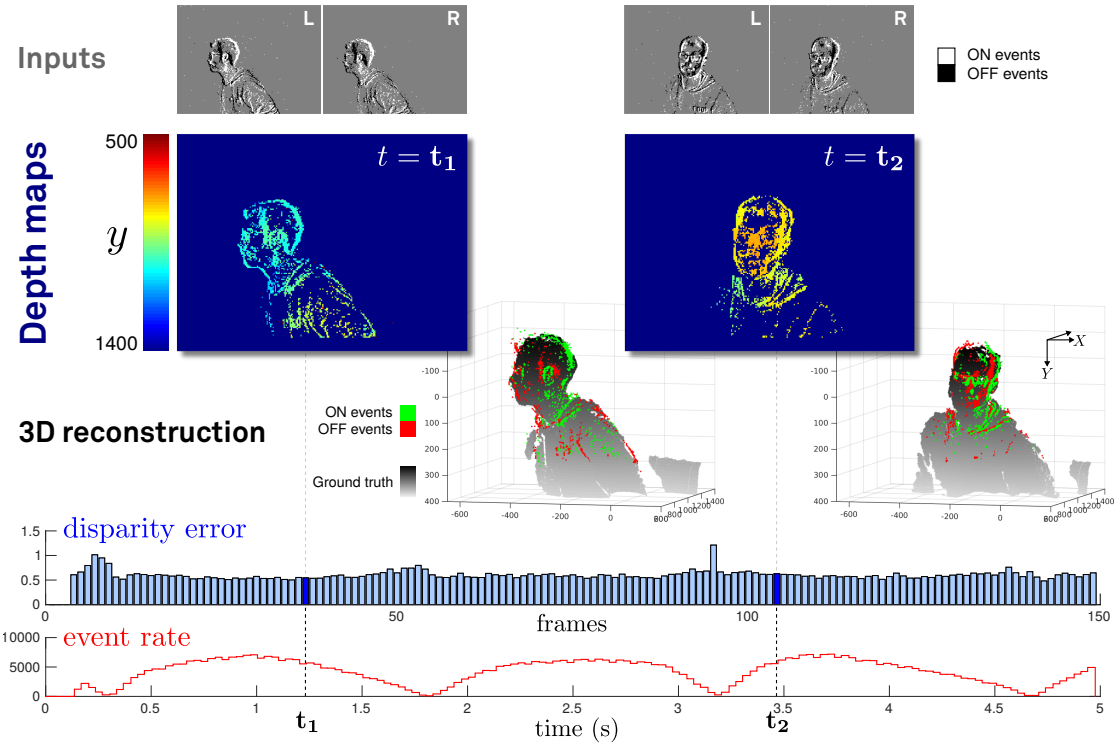


Figure 4.11: Qualitative and quantitative evaluation of the results from a natural scene in comparison to ground-truth data. This scene comprises a face that first rotates to the right (from its own point of view) followed by a rotation to the left, while initially moving towards the cameras before it rotates again rightwards and recedes. The scene had a duration of 5 seconds. On the bottom, the disparity error in pixels and input event rate are plotted over time. The images on the top show frames of input events which were accumulated at two arbitrarily chosen points in time, t_1 and t_2 , for each retina, labeled L (left) and R (right) accordingly. The output of the algorithm at time t_1 and t_2 is shown below in the form of depth maps which were obtained from accumulating 3D events. Depth (denoted by the coordinate y) is encoded in a color ranging from red (near) to blue (far). The bottom plots show point clouds of 3D events generated by triangulating matched events with their polarity, colored in green (ON) and red (OFF) accordingly. The ground-truth point cloud, which was obtained from the Kinect, is also shown (grey).

man et al., 2014). The local 2D correlation of time surfaces has been shown to be not only a metric for spatial correspondence (as is the case for normal images) but also a measure of the coherence of motion. The event-based STC stereo algorithm proposed here exploits this mechanism and has been demonstrated to provide a robust solution to complex stereo correspondence problems in dynamic scenes. Unfortunately, neither event-based datasets containing ground-truth values, nor a standardized evaluation procedure for event-based stereo algorithms exist. Therefore, two evaluation methods were developed. Usually, meaningful performance metrics for stereo algorithms are based on the disparity error (Scharstein and Szeliski, 2002) which requires a ground-truth disparity map. Here, the 3D error was computed by comparing reconstructed data with the associated ground-truth values, which were either modeled (see Section 4.3.2) or measured (see Section 4.3.3). An estimate of the disparity error was then obtained by back-projecting the 3D error. This approach had the disadvantage that

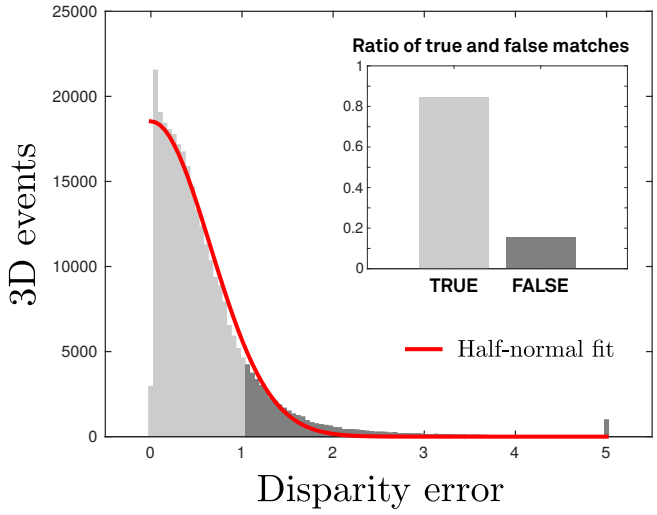


Figure 4.12: Disparity error histogram for the natural scene. Events corresponding to true matches (disparity error $\epsilon_d < 1$ pixel) are shaded light grey. All disparity errors that are greater than 5 pixels are contained in the last bin. A half-normal distribution was fitted to each of the histograms (red curve). The best fit yielded a mean disparity error of $\mu = 0.52$ pixels with a standard deviation of $\sigma = 0.65$. The inset shows the ratio of true and false matches.

it is only an approximation of the real disparity error. A better solution might have been to use a reference stereo camera system (Kogler et al., 2013). This method could be particularly interesting when using the DAVIS sensor because the grayscale images could be directly used to generate the ground-truth disparity maps, which would not require a calibrated reference system (event data and gray-scale images from the DAVIS are perfectly calibrated because they use the same pixels).

4.5.1 Area-based stereo algorithms revisited

The taxonomy for stereo algorithms introduced by Scharstein and Szeliski (2002) differentiates between local and global algorithms. Local algorithms compute disparity based on intensity values within a finite window at a given point. In contrast, global algorithms solve an optimization problem under explicit smoothness assumptions. Previously, local algorithms were referred to in this study as area-based algorithms. In Chapter 2, traditional examples of these kinds of algorithms were examined based on cost functions such as sum-of-squared-differences (SSD) or normalized-cross-correlation (NCC) (Hannah, 1974). The algorithm proposed in this study is basically a revision with one major difference: it applies to time surfaces instead of normal intensity images. Interestingly, it was observed that the NCC then produces a metric for spatial correspondence and motion coherence. In an extreme case where the areas have the exact same spatial content, the NCC metric computes the correlation of the velocities of the content. One of the challenges associated with area-based algorithms is choosing an appropriate window size. The literature suggests that the optimal window size is inversely proportional to spatial density and thus, it should vary (Marr, 1982; Banks et al.,

2004, 2005; Filippini and Banks, 2009). Generally, we can argue that larger windows provide more precision (a high certainty that the match is correct) but less accuracy (coarse disparity), whereas smaller windows provide less precision (a low certainty that the match is correct) with higher accuracy (fine disparity). A problem often associated with large areas is that spatial content is highly dense and the disparity gradients vary, meaning that correlation values are weak due to perspective distortion. In the experiments carried out during this study, however, only marginally decreased performance for relatively large windows was observed. This may be related to the presence of motion gradients, which add low spatial frequency content which improves the correlation (presuming that the motion is preferably fronto-parallel). Even in a case where the motion is not fronto-parallel, certain types of time surfaces — in this case the ETS — have been found to yield good results nevertheless. A possible explanation for this is given in Appendix A.3.3.

4.5.2 Efficiency of the event-based approach

It has been shown that temporal images and time surfaces can usefully represent visual information and can be directly applied to conventional algorithms, replacing classical images. While these unmodified algorithms still carry out the same operations, their underlying functional principle changes significantly and leads to the emergence of beneficial mechanisms. In the case of stereo vision, it has been observed that simple area-based approaches automatically extended their functional principal beyond ordinary spatial matching, to also encompass the matching of motion. The idea of extending a classical area-based algorithm to incorporate a further matching criterion based on motion might seem trivial. If classical images were used, however, this would need to be implemented differently, leading to a new algorithm. Conversely, when time surfaces are used, the unaltered algorithm can naturally implement the same matching criterion. While this simple event-based approach may be very elegant and interesting from a theoretical point of view, normal images (at a high frame rate) and a modified algorithm could achieve exactly the same results in practice. Therefore, the distinction from the classical frame-based approach is only interesting if the event-based approach is more efficient. Event-based approaches have the potential to be more efficient because event-based cameras produce non-redundant data. Does this imply that using time surfaces will automatically lead to a more efficient algorithm? To answer this question, the processing of time surfaces needs to be examined. The proposed algorithm computes the NCC of spatiotemporal features of temporally coarsely correlated events. This means that local areas of the time surfaces are processed at every occurrence of an event. Computing the NCC mainly consists of pairwise multiplication of the elements of the arrays which need to be correlated. In the case of time surfaces, the arrays are spatiotemporal features and their elements depend on the times at which events occur. Neighboring events with overlapping spatiotemporal features might lead to highly redundant computation as a result. Although the effective efficiency strongly depends on the scene content, it is clear from these considerations that the proposed algorithm is suboptimal and inefficient. Thus, how can the algorithm be implemented more efficiently? The following chapter is devoted to answering this question.

5 A Spiking Neural Network for Stereo Vision

5.1 Marr and Poggio revisited

The computational theory of human stereo vision proposed by Marr and Poggio (1976) presented a well-established framework, which introduced the concept of using cooperative processes to solve the stereo correspondence problem. Their work has made a huge impact in the field of stereopsis and machine stereo vision and has inspired the work presented here. When applying the key principles of the basic cooperative network to dynamic data in the form of temporal images, two important observations about the temporal dynamics are revealed. These observations ultimately led to the proposal of a spiking neural network for stereo correspondence.

5.1.1 Primary observation: Global support from coarse temporal correlation

In its original form, the algorithm formulated by Marr and Poggio (1976) cannot be applied to natural scenes comprising surfaces of varying depth. This scenario is illustrated in Fig. 5.1a. The two static retinal images show the edges of a plane that is slightly tilted in depth. The unit that represents the left edge $(4,4)$ does not receive excitatory input from the unit corresponding to the right edge $(6,7)$ because they do not have the same disparity, and thus, the network fails to suppress the false targets. Now, consider a biologically more plausible scenario in which the input does not consist of binary images but an array of retinal spiking neurons that encode temporal changes in illumination. In addition, the network's analog units are changed to spiking neurons and it is also assumed that the right edge of the tilted plane moves towards the left edge. For the moment, the excitatory and inhibitory connections are not considered. This scenario is illustrated in Fig. 5.1b. Time is encoded by shading, whereby more recent spikes are represented by darker gray values. The neurons in the network act as simple coincidence detectors that elicit a spike whenever both of their corresponding retinal neurons fire simultaneously within a short time window. Consequently, coincidence detectors at neighboring disparities of a moving target are also active. This is caused by coincidences between spikes from retinal neurons, which encode the actual position of the

target, and neurons from the other retina, which represent positions that the target recently passed. In other words, the sensitivity of the coincidence detector to coarse temporal delays produces supporting evidence at disparities where there is no actual target. Indeed, whereas the coincidence detectors at these disparities signal false targets, globally, they support true targets. This is achieved by means of the same mechanism described in Section 4.1.4, which describes the effect of asynchronous space-time sampling on the stereo correspondence problem. The beneficial effect here is that the false coincidences provide evidence of true distant targets. This is illustrated in Fig. 5.1b. Note how the right edge $(6,7)$ not only generates activity at its actual disparity ($d = 1$) but also at neighboring disparities ($d = 0, d = 2$). The analogy to the excitation along the line of equal disparity in the work of Marr and Poggio (1976) is now obtained by summing the evidence from coincidence detectors with equal disparity. Such integration is performed by an additional layer of disparity detectors. Thus, disparity detectors respond to a coherent pattern of coincidences, which provides an ideal signal of the true disparities. A detailed explanation will be provided in the following section. Fig. 5.1c and 5.1d illustrate how the networks may be extended to two-dimensional inputs, in which case each node has the depicted local structure. The two approaches are similar in that they both implement a cooperative mechanism that is only effective on the plane of fixed disparity. The nature of the inputs are significantly different. While the use of static images drastically limits the network's performance in scenes with varying depths, it can be observed that a similar network can overcome this shortcoming by using dynamic inputs and exploiting temporal correlations.

5.1.2 Profound observation: Local spatiotemporal correlation mechanism

Profound observations are obtained when studying the proposed modified network with two-dimensional temporal images. An illustration of this concept is shown in Fig. 5.2. Each layer is similar to the one-dimensional case illustrated in Fig. 5.1b. The times at which correlator units are active are encoded with grayscale values using the same notation as for temporal images. This means that more recent spikes are represented by darker intensities. To simplify the explanations which follow, the plane of *constant disparity* E_d and the plane of *constant horizontal cyclopean position* E_x are defined, whereby disparity is defined as $d = \bar{x}_R - \bar{x}_L$ and horizontal cyclopean position as $x = \bar{x}_R + \bar{x}_L$. Marr and Poggio (1976) observed that fronto-parallel spatial continuity induces activity in units located on the same line of disparity in the classical cooperative network. In the same way, in the proposed modified network, temporal continuity — which refers to objects that move in a fronto-parallel plane — produces activity even if the structure of the objects themselves is not fronto-parallel. One way of interpreting this is that the shortcoming of the traditional cooperative network is overcome by integrating motion cues. Moreover, it can be observed that activity on the plane E_x, which is perpendicular to E_d, indicates spatial and temporal non-conformity. This can be implemented as a further constraint to solve the correspondence problem. Figures 5.3 and 5.4 show two examples of how activity is distributed in the network when moving stimuli are truly and falsely matched. In the first example, the stimuli are tilted moving bars. The left part shows a true

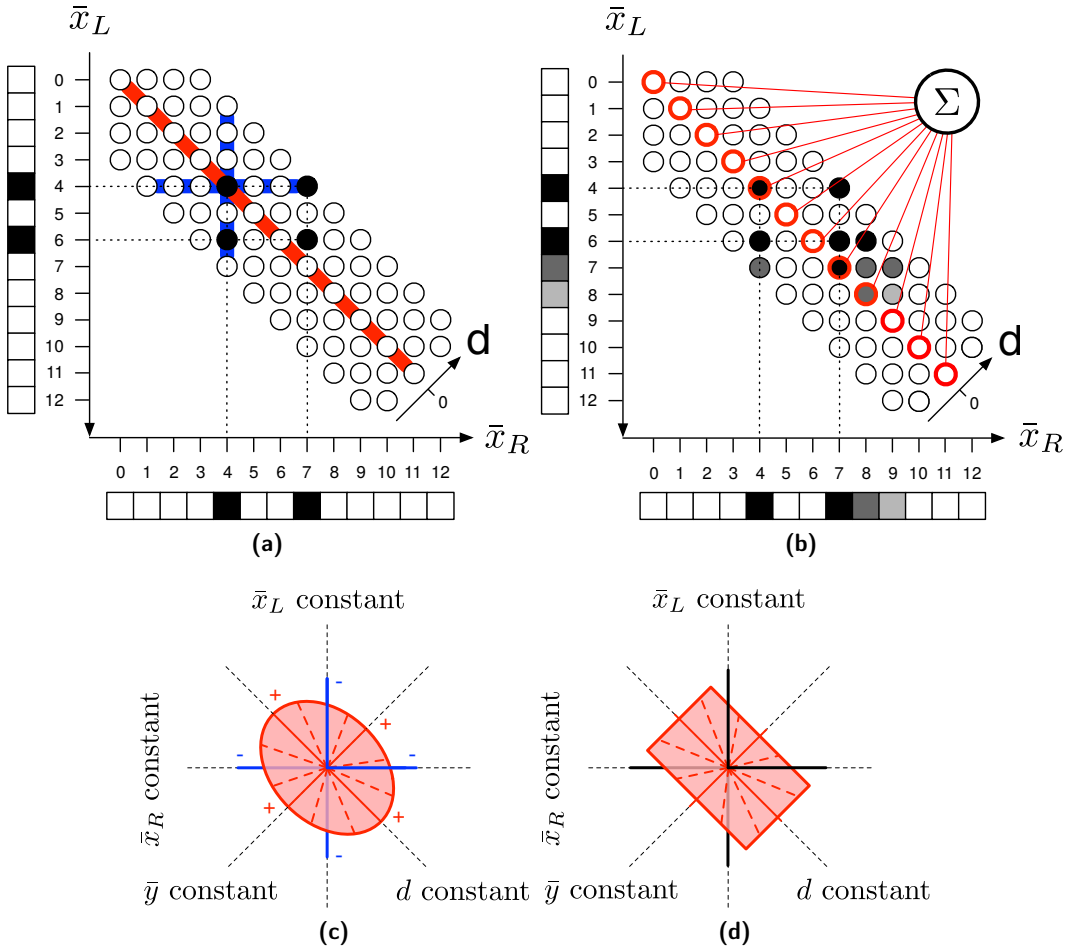


Figure 5.1: A comparison between the network from Marr and Poggio (1976) and the proposed modification that exploits temporal dynamics. **(a)** The original cooperative network for stereo correspondence by Marr and Poggio (1976) operates on static images. **(b)** The modified network with spiking units and dynamic inputs (here in the form of temporal images). **(c)** Extension to two-dimensional inputs for the original cooperative network. **(d)** Extension to two-dimensional inputs for the modified network.

match, whereby the bars have equal orientation and move in the same direction. Accordingly, there is only activity in E_d. In contrast, if the bars have different orientations but move in the same direction, activity is partially spread along E_x, and the match is considered false. Whether the match corresponds is determined by spatial compliance. Conversely, in the second example, temporal compliance (motion) identifies the correct match. In this case, the stimuli have equivalent spatial structure as the bars are vertically oriented. The true correspondence produces activity in E_d because of the coherent motion. Despite the fact that the spatial structures are the same, the false correspondence produces activity in E_x because the motion among the stimuli is dissimilar. Based on the work of Marr and Poggio (1976) it would already be expected that activity in E_d would somehow represent a measure

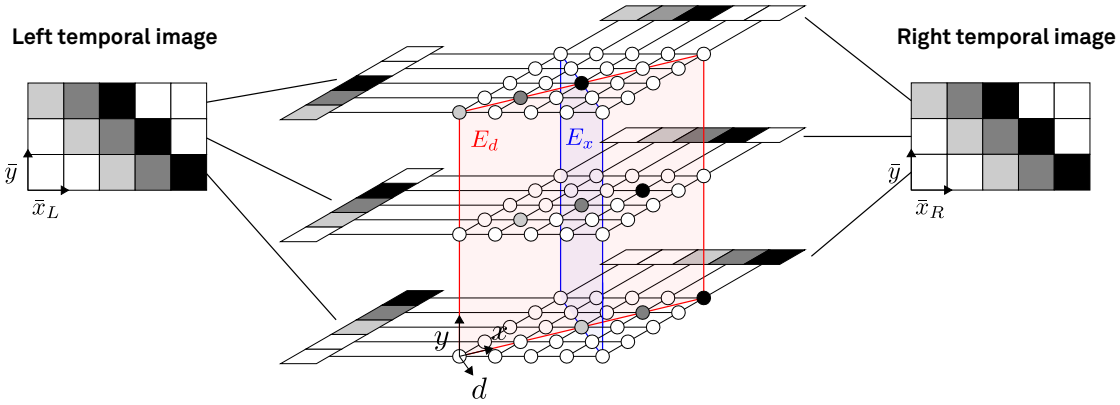


Figure 5.2: The cooperative stereo network revisited with temporal images. E_d and E_x indicate planes of constant disparity and constant horizontal cyclopean position respectively. In the example shown here, the visual stimulus consists of a tilted bar that moves from left to right.

of spatial correlation. Furthermore however, this example suggests that activity in E_d also encodes coherent motion (temporal compliance), whereas activity in E_x represents spatial anti-correlation or non-coherent motion.

5.1.3 Related Work

Hess (2006) should be credited for the original idea of an event-based stereo matching algorithm inspired by the cooperative network of Marr and Poggio (1976). Hess developed different types of disparity filters that could be applied to a matching matrix in order to measure temporal coincidences of events. Each element in the matching matrix had a representation in disparity space, similarly to the cooperative network. The disparity filters summed entries along the diagonals, which had the effect of favoring spatially and temporally smooth disparities. Contemporaneously to the project described here, other event-based stereo vision algorithms have been proposed based on the idea of cooperative networks (Piatkowska et al., 2013, 2014; Firouzi and Conradt, 2015). These approaches are different to the work in hand, however, in that they involve complex mechanisms such as temporal correlation kernels or dynamic weights. As will be shown, this work adopts a different approach, which builds upon very simple, yet effective mechanisms such as *spiking neurons*. As a result, while it is optimally suited to being implemented on neuromorphic hardware, it also retains a close link to neuroscience.

5.2 The spiking stereo neural network

In this chapter, a neural network capable of computing stereo correspondence is proposed, inspired by the approach of Marr and Poggio (1976). However, two major modifications of the original approach are adopted. Firstly, the input to the network does not comprise of static

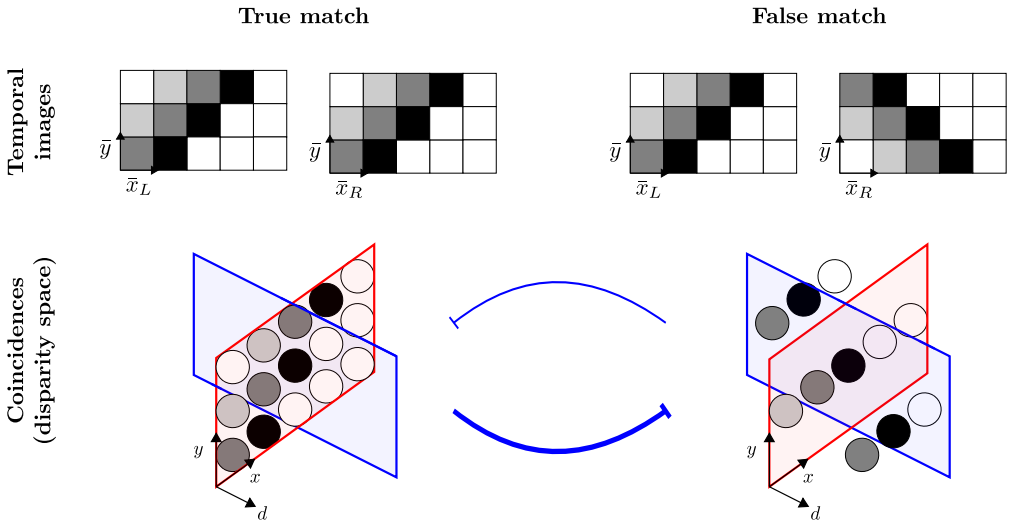


Figure 5.3: An example of network activity dominated by spatial correlation. Truly corresponding temporal images of moving tilted bars are shown on the right, whereas a pair of non-matching temporal images (of tilted bars with differing orientations) is shown on the left. Selected units of the network are shown and the planes E_d (red) and E_x (blue) are overlaid for reference. Whereas for the true match, activity occurs solely within E_d, in the second case, activity is partially shifted to E_x. If the network activity is integrated in a way such that units in E_d increase in sum, while units in E_x do the opposite, a measure of the correlation of the temporal images is obtained and the best match can be selected using winner-takes-all arbitration of all potential matches (illustrated by mutual inhibition between the true and false target).

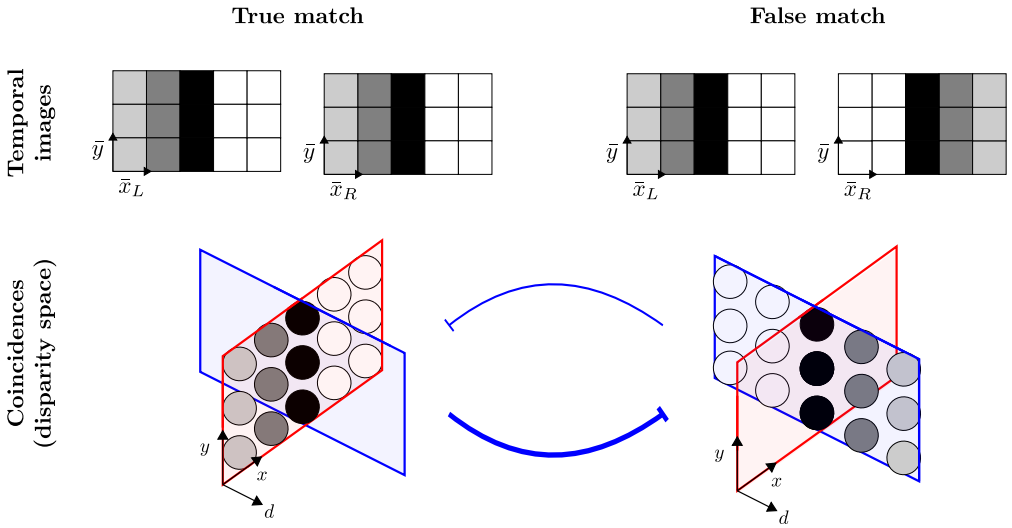


Figure 5.4: An example of network activity dominated by temporal correlation. Temporal images of two moving bars with the same orientation are shown. In the second case, however, they are moving in opposite directions . Similarly to the scenario illustrated in Fig. 5.3, the matching pair of temporal images produces more activity in E_d, whereas in the case of the non-matching pair, activity is shifted to E_x which results in a desired suppression of the response of the detector.

images, but of dynamic spatiotemporal visual information in the form of spike trains which are directly obtained from event-based, neuromorphic vision sensors. Secondly, the network comprises spiking neurons that operate en masse in a parallel, self-timed and data-driven fashion, similar to their real biological counterparts.

5.2.1 The coordinate system of the network

Similar to place cells in rats, each individual neuron in the network acts as a cognitive representation of a unique location in 3D space. The way in which this 3D space should be sampled by the neurons is not a trivial consideration. Consider a population of neurons for example, which represent uniformly distributed points in absolute-world coordinates. In this case, the way in which the retinas — here referred to as sensors — project into the network involves dynamic mapping, which is determined by the projective geometry of the sensor arrangement. Such mapping not only requires precise calibration, it also depends on the dynamic parameters of the system such as the focal length and convergence angle between the sensors. An alternative mapping technique could involve assigning a fixed pair of points, one from the image plane of each sensor, to each neuron. This neuron would then represent the spatial location of the point of intersection of the two lines of sight from its associated image points. Accordingly, the neuron would encode depth relative to the common fixation point of the sensors. The advantage of relating the neurons to the input rather than fixing their representation in space is that it allows for static mapping between the input and network, even if the sensor arrangement is varied. The exact process of this mapping mechanism and its implications for how the network samples the 3D space is explained as follows.

In Fig. 5.5a, a general stereo setup is shown comprising two sensors located at C_R and C_L and their image planes, each with its own image coordinates (x_R, y_R) and (x_L, y_L) respectively. The plane, spanned by any 3D point and the centers of the cameras is known as the *epipolar plane*. All points lying on an epipolar plane project to exactly one line in each image, the *epipolar line*. The epipolar lines are formed by the intersection of the epipolar plane with the image planes. Here, different epipolar planes are represented according to their inclination ϕ. The shaded blue plane indicates the horizontal epipolar plane as $\phi = 0$. In this example, five image points lying on the horizontal epipolar lines and their 25 correspondences in 3D space, forming a distorted array, are shown. Within this array, indices x and d along the diagonals are introduced. The indices are directly derived from the image coordinates:

$$\begin{aligned} x &= x_R + x_L \\ d &= x_R - x_L \end{aligned} \tag{5.1}$$

where x is referred to as the *horizontal cyclopean coordinate* and d as the *disparity coordinate*. Each epipolar plane now forms a layer of the network. Then, each neuron of the network can be uniquely described by the triplet (x, ϕ, d). In order to get an impression of how the representations of the neurons are distributed in 3D space, a different layer of the network,

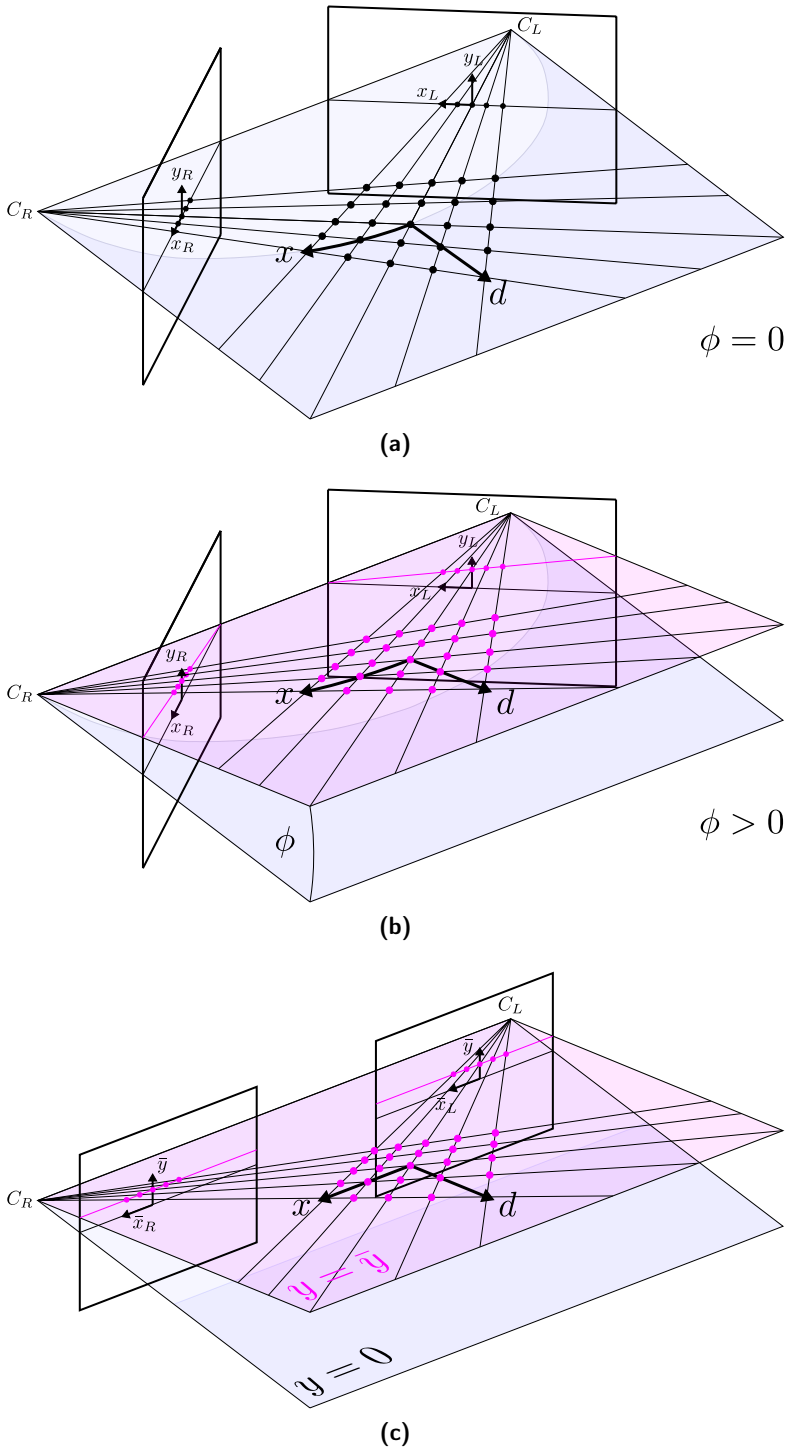


Figure 5.5: The coordinate system of the network and representation of disparity space. **(a)** The general sensor arrangement, showing the epipolar plane that is perpendicular to the image planes. **(b)** The same sensor arrangement, showing an inclined epipolar plane. **(c)** The same sensor arrangement with rectified image planes.

corresponding to an inclined epipolar plane ($\phi > 0$) is shown in Fig. 5.5b. The challenge that remains is to define the relationship between the image coordinates and the inclination ϕ of the epipolar plane, as this depends on the poses of the sensors. This problem is resolved using the concept of image rectification, which is graphically illustrated in Fig. 5.5c. Image rectification is a homographic transformation that re-projects the image planes so that they are coplanar. The image points are then expressed in *rectified coordinates*, $(\bar{x}_R, \bar{y}_R)$ and $(\bar{x}_L, \bar{y}_L)$. Rectification requires a number of correspondences among the points and is a standard procedure in machine vision. Rectified coordinates entail corresponding epipolar lines which have the same constant vertical coordinate $\bar{y}_L = \bar{y}_R = \bar{y}$. The inclination of the epipolar plane is directly related to $\bar{y}$ and provides a better representation of disparity space. Thus, a new representation is obtained by using rectified image coordinates and redefining $d = \bar{x}_R - \bar{x}_L$, $x = \bar{x}_R + \bar{x}_L$ and $y = \bar{y}$ accordingly. A unique map $\mathscr{M}$ can then be derived, which is invariant to sensor pose, and which assigns a neuron to each epipolar pair of rectified image coordinates. This neuron can be described by the triplet (x, y, d):

$$\begin{array}{rccl} \mathscr{M}: & \mathbb{N}^3 & \rightarrow & \mathbb{D}^3 \\ & (\bar{x}_L, \bar{x}_R, \bar{y}) & \mapsto & (x, y, d) = (\bar{x}_R + \bar{x}_L, \bar{y}, \bar{x}_R - \bar{x}_L) \end{array} \tag{5.2}$$

where (x, y, d) are the network coordinates and their range is the *disparity space* $\mathbb{D}^3$. Finally, each neuron is uniquely assigned a horizontal and vertical cyclopean coordinate x and y, as well as a disparity coordinate d. Together, these coordinates represent a point in disparity space $\mathbb{D}^3$, which corresponds to the neuron's cognitive representation of a location in 3D space. The absolute-world coordinates of this location are determined by the intersection of the lines of sight from the pair of rectified image points, which can be derived from the network coordinates by means of the inverse mapping function $\mathscr{M}^{-1}$.

5.2.2 The architecture of the network

A spiking neural network is proposed that extends the classical cooperative network by adopting the previously described mechanism of stereo correspondence based on temporal and spatial compliance. An abstract view of the entire network architecture is given in Fig. 5.6a. The retinal cells (or pixels of the sensors) are represented by the populations L and R and serve as the input to the network. Their size is indicated by n^2, as the sensors consist of two-dimensional pixel arrays. L and R excite a population of neurons C, referred to as the "coincidence detectors". The size of the population C scales cubically because the neurons within it encode coincidences that occur in disparity space. Lastly, another population of neurons D, termed the "disparity detectors", pool responses from C in mixed excitatory and inhibitory manner. In order to signal only correct disparities, the recurrent inhibitory connections among neurons in D implement a winner-takes-all mechanism designed to suppress disparity responses to false targets. A detailed view of a horizontal layer of the network is illustrated in Fig. 5.6b. Following on from the rationale for using temporal images, spiking

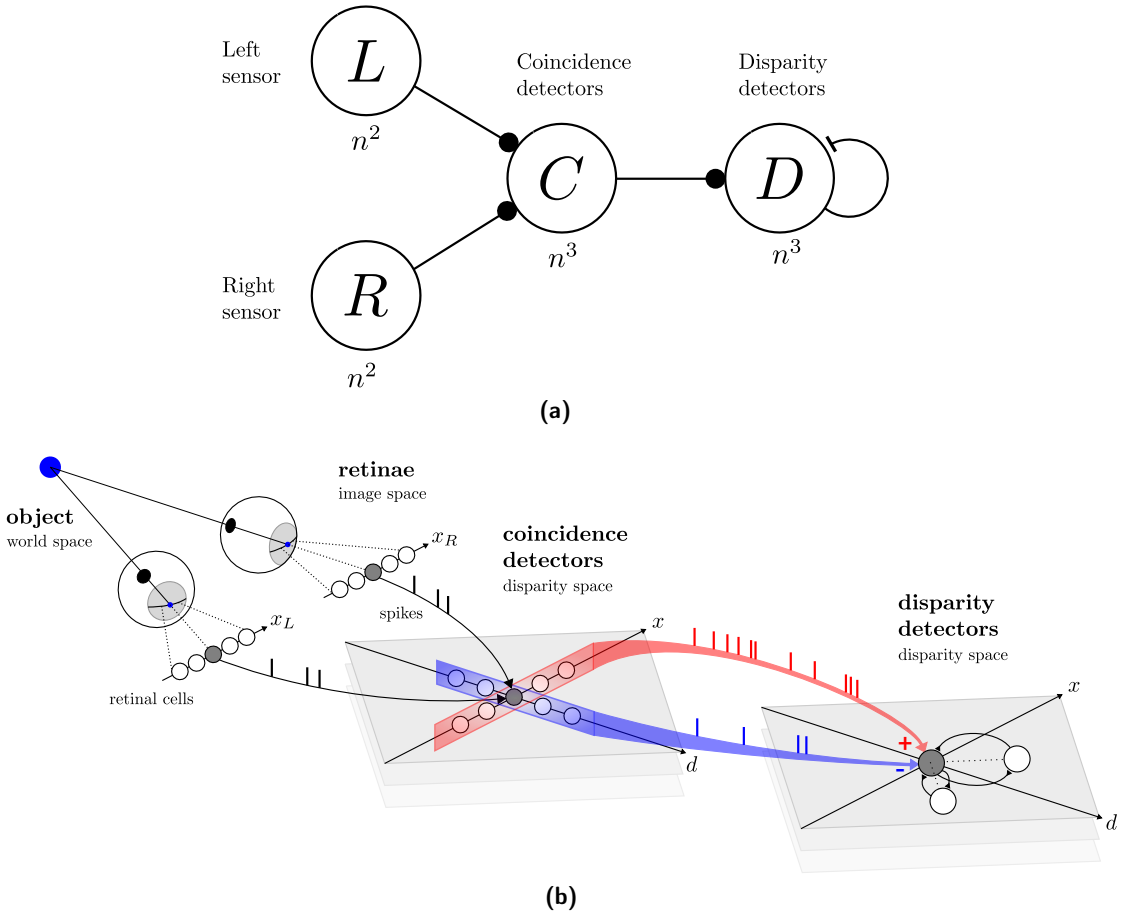


Figure 5.6: The spiking stereo network. **(a)** Abstract view of the network's architecture. **(b)** Detailed view of a horizontal layer of the network. An object is sensed by two eyes and accordingly projected onto their retinal cells. The spiking output of these cells is spatiotemporally correlated (coincidence detectors) and integrated (disparity detectors). The final output encodes a representation of the original scene in disparity space. For the sake of visibility, only a horizontal line of retinal cells, at fixed vertical cyclopean position y, is considered. The corresponding coincidence and disparity detector units, hence, lie within a horizontal plane. Again, only a few units are shown here whereas in the complete network, the units would be uniformly distributed over the entire plane. The shaded planes indicate how the network expands vertically over y.

retinal cells are used as inputs. Each spike represents a change in illumination at a specific spatial position at a particular time. For each pair of corresponding horizontal lines of retinal cells from L and R, a horizontal layer of neurons in C signals temporally coinciding spikes. The cells in C are arranged according to the previously defined disparity coordinate system of the network. Thus, each cell has a unique spatial representation in disparity space (x, y, d) (only x and d are shown) such that each spike provides evidence for a potential target at the corresponding real-world disparity position. Thus, the complete population of coincidence detectors encodes all potential targets (true and false disparities). The disparity detectors implement a binocular correlation mechanism, which is realized by integrating the responses from coincidence detectors within the planes of constant disparity E_d and cyclopean position

E_x. Activity in E_d constitutes supporting evidence for true matches (excitation of disparity detector), whereas activity in E_x denotes countervailing evidence (inhibition of disparity detector). Finally, the uniqueness constraint is enforced by mutual inhibition of disparity detectors that represent spatial locations in the same line of sight.

5.2.3 Simple coincidence detectors

In order to implement a neural coincidence detection mechanism, the proposed network uses neurons with leaky-integrate-and-fire (LIF) dynamics (Gerstner and Kistler, 2002). The membrane potential $v_{\mathbf{c}}(t)$ of a LIF coincidence neuron is described by the following equation

$$\begin{cases} \tau_{\mathbf{c}} \dfrac{\mathrm{d}v_{\mathbf{c}}(t)}{\mathrm{dt}} = -v_{\mathbf{c}}(t) + I_{\mathbf{c}}(t)\,, & v_{\mathbf{c}}(t) < \theta_{\mathbf{c}} \\ v_{\mathbf{c}}(t) = 0\,, & v_{\mathbf{c}}(t) \geq \theta_{\mathbf{c}} \end{cases} \tag{5.3}$$

where the time constant $\tau_{\mathbf{c}}$ determines the neuron's leak and $\theta_{\mathbf{c}}$ the threshold at which the neuron fires. A coincidence neuron receives input from a pair of epipolar retinal cells, which can be described as a sum of spikes

$$I_{\mathbf{c}}(t) = w \sum_i \delta_{\bar{x}_L}(t - t_i) + w \sum_j \delta_{\bar{x}_R}(t - t_j) \quad \Big| \quad \mathbf{c} = \mathscr{M}(\bar{x}_L, \bar{x}_R, \bar{y}) \tag{5.4}$$

where the indices i and j indicate the spike times of the retinal cells $(\bar{x}_L, \bar{y})$ and $(\bar{x}_R, \bar{y})$ respectively. A single spike is modeled with the Dirac function $\delta(t)$. For the obvious reason of symmetry, the synaptic weights w are equally sized for both inputs. The subscript vector $\mathbf{c} = (x_c, y_c, d_c)$ corresponds to the neuron's unique spatial representation in disparity space, explicitly defined by the previously introduced map $\mathscr{M}$. If it is assumed that spikes from the same retinal cell are temporally well separated, such that a preceding spike only has a marginal effect on the membrane potential at the time of a current spike, then the neuron's *sensitivity* $S_{\Delta T}$ to interocular temporal delays can be derived as:

$$S_{\Delta T} = \tau_{\mathbf{c}} \ln\left(\frac{1}{\frac{\theta_{\mathbf{c}}}{w} - 1}\right), \quad 1 < \frac{\theta_{\mathbf{c}}}{w} \leq 2 \tag{5.5}$$

The sensitivity of a coincidence detector is the range of interocular temporal delays between two spikes from an epipolar pair of retinal cells within which the neuron is responsive. The ratio $\frac{\theta_{\mathbf{c}}}{w}$ is chosen so that while two retinal spikes can trigger a response, a single spike will not. It is difficult to select an appropriate time constant $\tau_{\mathbf{c}}$ because it directly affects the sensitivity of the neuron. If the sensitivity is too high, the neuron will be selective to long interocular temporal delays, which will increase the number of coincidences associated with ambiguous disparities. On the other hand, if the sensitivity is too low, the beneficial effect of

global support from distant targets with varying disparity will be reduced (see Section 5.1.1).

5.2.4 Complex disparity detectors

Similarly to complex cells in the brain, the proposed disparity detectors aggregate evidence from the responses of simple coincidence detectors. The disparity detectors are also modeled using LIF neuron dynamics, but with a distinct time constant $\tau_{\mathbf{d}}$ and a firing threshold $\theta_{\mathbf{d}}$:

$$\begin{cases} \tau_{\mathbf{d}} \dfrac{\mathrm{d}v_{\mathbf{d}}(t)}{\mathrm{d}t} = -v_{\mathbf{d}}(t) + I_{\mathbf{d}}(t)\,, & v_{\mathbf{d}}(t) < \theta_{\mathbf{d}} \\ v_{\mathbf{d}}(t) = 0\,, & v_{\mathbf{d}}(t) \geq \theta_{\mathbf{d}} \end{cases} \tag{5.6}$$

The input of the disparity detector at $\mathbf{d} = (x_d, y_d, d_d)$ combines the outputs from coincidence detectors within bounded planar excitatory and inhibitory regions in disparity space $C^+ \in \mathbb{D}^2$ and $C^- \in \mathbb{D}^2$ respectively:

$$I_{\mathbf{d}}(t) = w_{exc} \sum_{\mathbf{c} \in C^+} \sum_k \delta_{\mathbf{c}}(t - t_k) - w_{inh} \sum_{\mathbf{c} \in C^-} \sum_k \delta_{\mathbf{c}}(t - t_k) \tag{5.7}$$

where k represents the index of the spike times of coincidence neuron $\mathbf{c}$, while w_{exc} and w_{inh} are constant excitatory and inhibitory weights. The regions C^+ and C^- are squared windows in the plane of constant disparity E_d and the plane of constant horizontal cyclopean position E_x, which are defined relative to the disparity detector's spatial representation $\mathbf{d}$:

$$C^+ = \left\{ \mathbf{c} \in C \,\middle|\, (|x_c - x_d| \leq \omega) \wedge (|y_c - y_d| \leq \omega) \wedge (d_c = d_d) \right\} \tag{5.8}$$

$$C^- = \left\{ \mathbf{c} \in C \,\middle|\, (x_c = x_d) \wedge (|y_c - y_d| \leq \omega) \wedge (|d_c - d_d| \leq \omega) \right\} \tag{5.9}$$

where ω is half of the window size. The synaptic weights should be chosen such that they are inversely proportional to the sizes of C^+ and C^-. Since C^+ and C^- are of equal size, it is suggested that $w_{exc} = w_{inh}$. The time constant of the disparity neurons determines how evidence from the past is weighted, similarly to the decay constant of the time surfaces which were previously introduced (see Section 4.2.2). Making an appropriate choice of $\tau_{\mathbf{d}}$ is dependent on the stimuli but in general, it is significantly larger than $\tau_{\mathbf{c}}$.

Variants of disparity detectors

Ideally, the proposed disparity detectors would compute the spatiotemporal cross-correlation of interocular temporal images. However, cross-correlation is a complex computation that involves the calculation of the covariance followed by normalization. The responses of the

proposed disparity detectors are clearly not normalized, which can be easily observed by comparing the case of a single coincidence in C^+ with the case of multiple coincidences in C^+. Assuming there are no coincidences in C^- in both cases, the latter results in a stronger response while the cross-correlation coefficient would be ideal ($\rho = 1.0$) in both scenarios. For the purposes of this project, it is argued that it is sufficient for the proposed disparity detectors to compute a covariance-like measure, while the problem of non-normalized responses is addressed by mutual inhibition (see next section). To study the computation performed by the disparity neuron, various types of detectors have been examined. Three types of complex disparity detectors are proposed, each characterized by their receptive fields:

$$\mathrm{RF_I} = C^+ \wedge C^- \tag{5.10}$$

$$\mathrm{RF_{II}} = C^+ \tag{5.11}$$

$$\mathrm{RF_{III}} = C^+ \wedge L^- \wedge R^- \tag{5.12}$$

The receptive fields of the detectors of type I and II are illustrated in Fig. 5.7. The third type of disparity detector combines the excitatory input from C^+ with two inhibitory regions, L^- and R^-, which are directly located in the populations of retinal cells. Epipolar retinal cells are indexed with the vectors $\mathbf{p} = (\bar{x}_L, \bar{y})$ and $\mathbf{q} = (\bar{x}_R, \bar{y})$ respectively:

$$L^- = \left\{ \mathbf{p} \in L \,\middle|\, (|\bar{x}_L + \bar{x}_R - x_d| \leq \omega) \wedge (|\bar{y} - y_d| \leq \omega) \right\} \tag{5.13}$$

$$R^- = \left\{ \mathbf{q} \in R \,\middle|\, (|\bar{x}_R + \bar{x}_L - x_d| \leq \omega) \wedge (|\bar{y} - y_d| \leq \omega) \right\} \tag{5.14}$$

The input of detector type III can thus be adapted as follows:

$$I_{\mathbf{d}}(t) = w_{exc} \sum_{\mathbf{c} \in C^+} \sum_k \delta_{\mathbf{c}}(t - t_k) - w_{inh} \left(\sum_{\mathbf{p} \in L^-} \sum_i \delta_{\bar{x}_L}(t - t_i) + \sum_{\mathbf{q} \in R^-} \sum_j \delta_{\bar{x}_R}(t - t_j) \right) \tag{5.15}$$

A detailed view of the local structure of a complex disparity detector of type III is illustrated in Fig. 5.8.

5.2.5 Mutual inhibition of disparity detectors

As the disparity detectors are not expected to address the problem of normalization, a fixed threshold $\theta_{\mathbf{d}}$ will result in one of two problems. If the threshold is low, disparity detectors are more likely to respond to false but similar targets, particularly at locations where the targets are large or rapidly moving (many coincidences), even in the presence of inhibiting evidence. In contrast, if the threshold is high, although the sensitivity to false disparities is

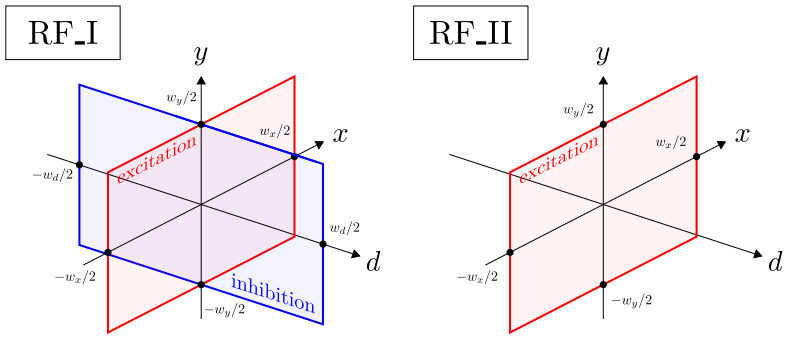


Figure 5.7: Receptive fields of two types of disparity detectors with (RF_I) and without inhibitory zone (RF_{II}).

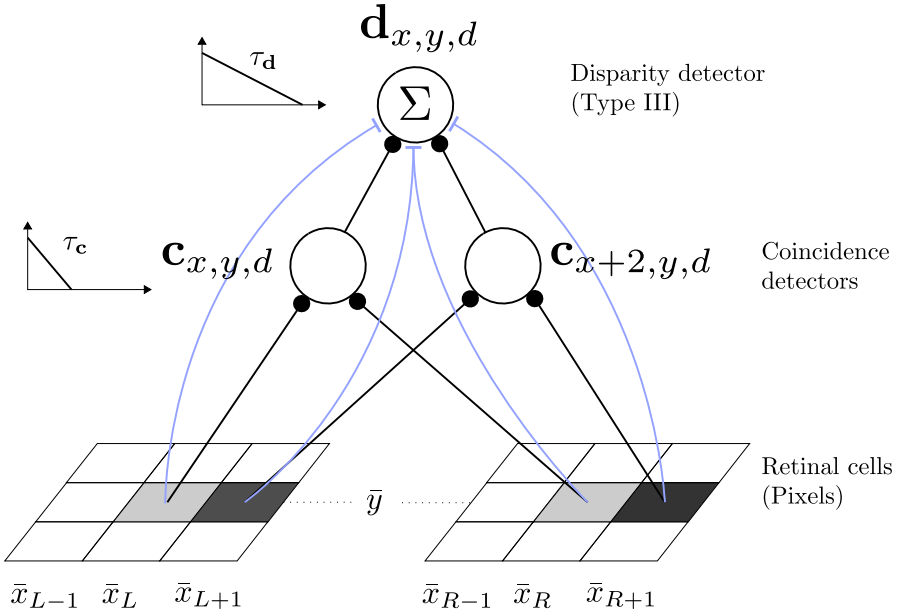


Figure 5.8: Local structure of a complex disparity detector of type III. The region of excitatory inputs from coincidence detectors is equivalent to that of types I and II, but inhibitory inputs arrive directly from retinal cells.

reduced, the response to true disparities associated with small or slowly moving targets is also diminished (fewer coincidences). This suggests that $\theta_{\mathbf{d}}$ is a context-dependent and critical parameter. The proposed network addresses this problem by implementing the uniqueness constraint first described by Marr and Poggio (1976). Given a disparity detector which is spatially represented at the position of a false disparity, the correct disparity is located somewhere along the line of sight and is simultaneously represented by another neuron. Ideally, this neuron integrates more coinciding evidence, thus evoking a faster response. This response can then be recurrently fed as an inhibitory input into the neuron located at the false disparity in order to suppress its response. On a macro level, this concept results in winner-take-all synchronization among neurons on the same line of sight. In the proposed network, this mechanism is implemented with mutual inhibition, such that each disparity detector inhibits all the other neurons in its line of sight. Thus, Eq. 5.6 needs to be updated by subtracting an inhibitory current $I_{\mathbf{d}-}(t)$ accordingly:

$$\begin{cases} \tau_{\mathbf{d}} \dfrac{\mathrm{d}v_{\mathbf{d}}(t)}{\mathrm{dt}} = -v_{\mathbf{d}}(t) + I_{\mathbf{d}}(t) - I_{\mathbf{d}-}(t)\,, & v_{\mathbf{d}}(t) < \theta_{\mathbf{d}} \\ v_{\mathbf{d}}(t) = 0\,, & v_{\mathbf{d}}(t) \geq \theta_{\mathbf{d}} \vee v_{\mathbf{d}}(t) \leq 0 \end{cases} \tag{5.16}$$

whereby the inhibitory current is defined as

$$I_{\mathbf{d}-}(t) = w_{rec} \sum_{\mathbf{d} \in D^-} \sum_{n} \delta_{\mathbf{d}}(t - t_n) \tag{5.17}$$

and the inhibitory region D^- is defined by the two lines of sight (one from each retina):

$$D^- = \left\{ \mathbf{d} \in D \;\middle|\; (x_d - d_d = 2\bar{x}_L) \vee (x_d + d_d = 2\bar{x}_R) \right\} \tag{5.18}$$

If the recurrent synaptic weights are set such that $w_{rec} \geq \theta_{\mathbf{d}}$, then each spike that signals a correct disparity will completely reset the membrane potential of the other neurons in D^-. This is preferable as it prevents neurons at false disparities from gaining a head start when competing to signal the next correct disparity directly after a spike. In order to avoid a negative bias, a further condition is added to Eq. 5.16, that does not admit potentials below zero.

5.2.6 Representation and coding of disparity

If a disparity detector is placed at each spatial location in $\mathbb{D}^3$, neighboring neurons have strongly overlapping receptive fields. A natural consequence is that disparity detectors also respond to targets which are at their preferred disparity, but distant from their preferred cyclopean position. This behavior corresponds exactly to real disparity-tuned complex cells, which are known to be position invariant. This means that such cells respond equally to any target within their receptive fields, regardless of its exact position. Thus, a homogeneously

distributed population of disparity detectors in $\mathbb{D}^3$ produces redundant responses. Therefore, a broader spacing in the direction of the cyclopean coordinates (but not in the direction of disparity) is suitable. If the output of the network is considered, the entirety of all disparity spikes would correspond to a blurry dynamic disparity map, as disparity detectors are broadly tuned to cyclopean position. However, an accurate disparity map can be obtained by combining the output from coincidence and disparity detectors. Recall that coincidence detectors encode the entirety of all possible disparities (corresponding to true and false targets). Thus, if a disparity neuron spikes, the simultaneous responses of coincidence detectors that have an equal (or nearby) spatial representation in $\mathbb{D}^3$ reveal the correct and accurate disparities. Once again, the notion of events is useful to express the disparity response of the network. A *unipolar disparity event* $e_d^+ = (\mathbf{d}, t)$ represents a relative change in light intensity, that occurred at time t, at location $\mathbf{d}$, in $\mathbb{D}^3$. Similarly, a *unipolar coincidence event* $e_c^+ = (\mathbf{c}, t)$ represents coinciding evidence of a change in light intensity at location $\mathbf{c}$ in $\mathbb{D}^3$ at time t. The entirety of all coincidence and disparity events are defined by the sets $\mathcal{C}^+ = \{ e_c^+ \}$ and $\mathcal{D}^+ = \{ e_d^+ \}$ respectively. The final output of the network that is obtained from the combination of coincidence and disparity responses is then defined as the set of *filtered unipolar disparity events*:

$$\mathcal{O}^+ = \mathcal{C}^+ \cap \mathcal{D}^+ \tag{5.19}$$

The set $\mathcal{O}^+$ encodes spatially precise and unambiguous light intensity changes in disparity space. The network is extended by also considering the sign of intensity change, which is encoded using an additional event attribute, the polarity $s \in [-1, 1]$. In order to handle the polarity properly, a pair of coincidence detectors, one for each polarity, is used at each location in disparity space, similar to the simple ON and OFF cells in V1. Thus, the definition of coincidence events is modified accordingly: $e_c = (\mathbf{c}, s, t)$. Disparity detectors give equal weight to the evidence from both types of coincidence detectors. Thus, the polarity of the disparity detectors themselves is neutral. However, the filtered disparity events inherit the polarity of the coincidence events, which results in a set $\mathcal{O}$, that encodes spatially precise and unambiguous, polarized light intensity changes in disparity space:

$$\mathcal{O} = \left\{ e_c \in \mathcal{C} \,\middle|\, [e_c]^+ \in \mathcal{D}^+ \right\} \tag{5.20}$$

where $[]^+$ denotes a polarity rectification that transforms the polarity event $e_c = (\mathbf{c}, s, t)$ to a unipolar event $e_c = (\mathbf{c}, t)$.

5.3 Experiments and Results

5.3.1 Spatiotemporal correlation mechanism of complex disparity detectors

Three different types of disparity detectors were proposed, which aim to approximate a sort of local correlation of spatiotemporal visual input in the form of spike events that encode temporal contrast changes. The proposed models do not account for normalized responses. Thus, it was suggested that the underlying mathematical computation is better described by the covariance rather than a normalized form of cross-correlation (e.g. NCC). In the following experiment, the responses of the three types of disparity detectors were recorded and compared to the result of the covariance computation. Fig. 5.9 illustrates the experimental procedure. The spiking visual input is represented in the form of *bipolar temporal images*, which are temporal images that also encode the sign of the intensity changes (polarity). To obtain a bipolar temporal image, a temporal image is multiplied one element at a time by the polarity of the last event at each spatial position. For this experiment, the images were cropped to match the receptive field size of the disparity neuron. Images were extracted in pairs from the left and right visual source at the locations of temporally coinciding events with epipolar spatial positions. For each such pair, regardless of whether it corresponded to a true or false match, the spike times were fed into a network. This network employed a single disparity detector and its response was recorded as the value of the membrane potential $v_{\mathbf{d}}$ (the threshold non-linearity was removed). Simultaneously, the covariance of the image pair was computed as

$$\text{cov} = \sum_{\mathbf{p}} \left(\Sigma_L^+(\mathbf{p}) - \mu_{\Sigma_L^+} \right) \left(\Sigma_R^+(\mathbf{p}) - \mu_{\Sigma_R^+} \right) \tag{5.21}$$

where Σ_L^+ and Σ_R^+ are the bipolar temporal images and $\mu_{\Sigma_L^+}$ and $\mu_{\Sigma_R^+}$ are their associated mean values.

Type III disparity detectors

Disparity detectors of type III have the most complex structure as they combine the inputs from coincidence neurons and retinal cells. This kind of detector was tested on a dynamic scene that portrays a face nodding from side to side. The scene was recorded with two ATIS sensors. The scene lasted 3 seconds and about 2 million coincidences (potential matches) were investigated. In this experiment, only the best match was considered among the temporal coincidences on the same epipolar line. This resulted in a total of 314,622 recorded data points. A data point consists of the neural response of the disparity detector and the value of the covariance (Eq. 5.21) for the matching pair of bipolar temporal images. A sparse random selection of the recorded data points is shown in Fig. 5.10. The color map encodes the vertical position y in disparity space of the coinciding events. Two clusters that highly depend on y are observed. The clustering can be explained by the composition of the scene. The nodding

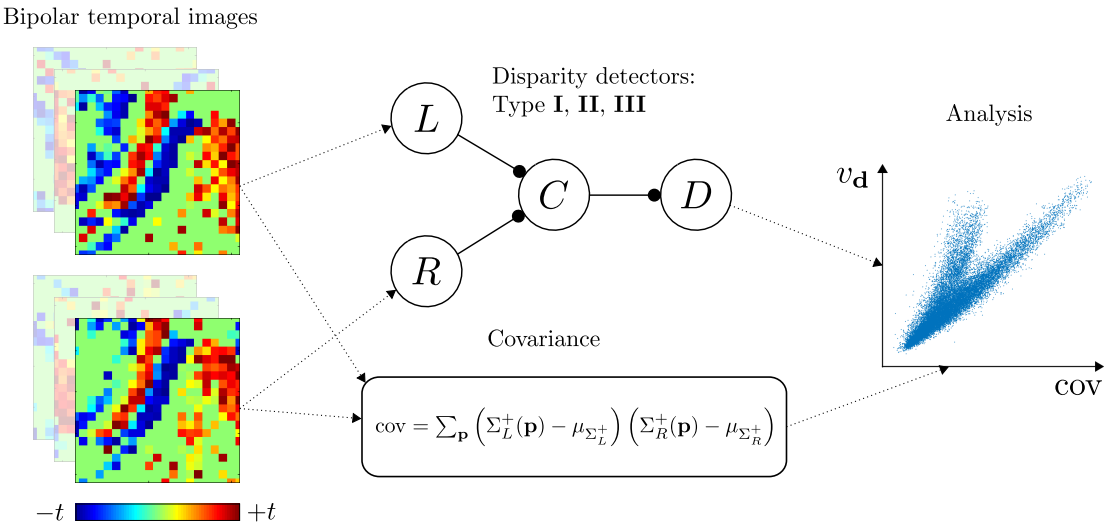


Figure 5.9: Schematic of the experimental procedure to investigate the underlying computation of the proposed disparity detectors.

from side to side causes the upper part of the face (larger y) to move fast while the lower part including the neck and shoulders (smaller y) nearly remain at the same position. This suggests that the neural response has a more distinctive dependency on stimuli velocity than the covariance. However, within the individual clusters, there seems to be a strongly correlating and closely linear relationship between the neural response and the covariance. The Pearson's correlation coefficient (PCC) of the two variables was found to be $\rho = 0.95$ which is strong evidence for similarity. However, it should be considered that due to the way the data points were selected, a bias towards better matching pairs of bipolar images was introduced. Therefore, no claim about the relationship between the neural response and covariance of false targets can be made.

To assess the neuron's behavior in the presence of false targets, three regions of interest (ROI) in disparity space have been identified. These regions are dominated by *fast moving true targets* (TT^+), *slowly moving true targets* (TT^-) and *false targets* (FT). Fig. 5.11a shows a 3D view of the disparity events which were matched in the scene with the three ROIs highlighted accordingly. The disparity events were obtained using two different methods. The first method is identical to the event-based STC stereo algorithm that was described in Section 4.2.6, whereby correct matches are selected based on the highest NCC of spatiotemporal features of temporarily coinciding epipolar events. The second method is a slight modification of the first, termed the *neural covariance approximation* (NCA) of the STC algorithm, whereby the matching cost corresponds to the response (membrane potential $v_{\mathbf{d}}$) of the proposed disparity detectors (here type III). The size of an ROI is defined as the total number of disparity events that occur within the ROI. Accordingly, the ROI sizes were determined for both methods while changing the matching cost threshold. This threshold sets the minimum cost that is required for a match to be considered true and produce a disparity event. The normalized ROI sizes are

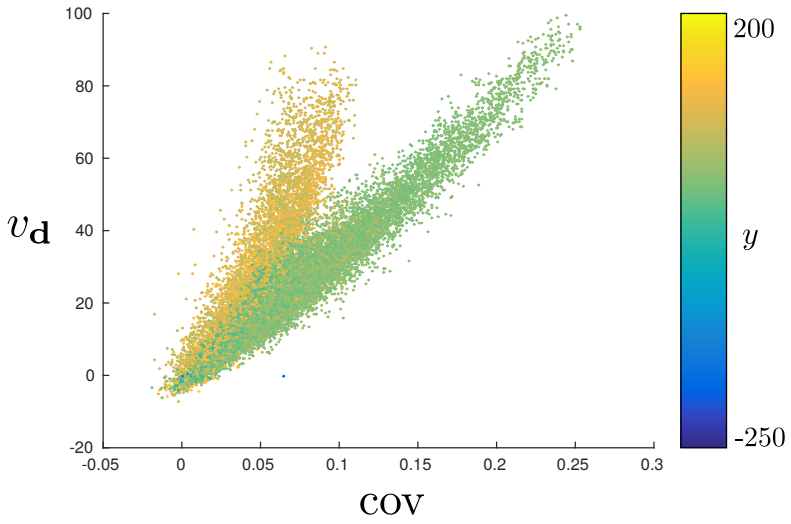


Figure 5.10: Comparison of the neural response of disparity detectors of type III ($v_{\mathbf{d}}$) and covariance (cov) of spatiotemporal visual stimuli. The data shows a dependency on stimuli velocity which is related to the vertical position y in this particular scene. A group of fast moving stimuli (yellow) were located at the top (large y) and another group of slow moving stimuli (green) at the bottom (low y).

obtained through division by the maximum value, which corresponds to the ROI size obtained when the threshold value is zero. The normalized ROI sizes, $\bar{n}_{\text{STC}}$ and $\bar{n}_{\text{NCA}}$, for both methods are plotted against each other as shown in Fig. 5.11b. Each data point shows corresponding $\bar{n}_{\text{STC}}$ and $\bar{n}_{\text{NCA}}$ for a common relative threshold that represents the ratio of accepted events p. This ratio is defined as the number of (accepted) disparity events produced (where the matching cost was higher than the individual thresholds) divided by the maximum number of disparity events (the case where the individual thresholds are minimal). In this manner, the leftmost data point corresponds to an acceptance rate of $p = 100\%$, while it linearly decreases with every further point until the minimum $p = 0\%$, at the rightmost data point. If a data point is located in the gray region above the diagonal, this indicates that the NCA method is more sensitive to targets in the specified ROI ($\bar{n}_{\text{NCA}} > \bar{n}_{\text{STC}}$) at the given threshold. At first, it is observed that both methods are equally selective for fast moving true targets (TT^{+}) over the entire range of varying thresholds. However, for slowly moving true targets (TT^{-}) the STC method shows stronger selectivity at increased thresholds, while being less sensitive to false targets (FT). This suggests that increasing the matching threshold in the case of the STC algorithm is a suitable strategy to reduce ambiguous disparities. However, in the case of the NCA algorithm, this also reduces the sensitivity to slow targets, making the method strongly dependent on the context. A qualitative comparison of the output of the two algorithms is shown in Fig. 5.12 for three different thresholds. It can be observed that the STC methods conserve weak targets in the TT^{-} ROI, even for high acceptance rates ($p = 25\%$). However, one advantage of the NCA algorithm is that it is not selective to coinciding noise events, as there is no supporting evidence in their vicinity.

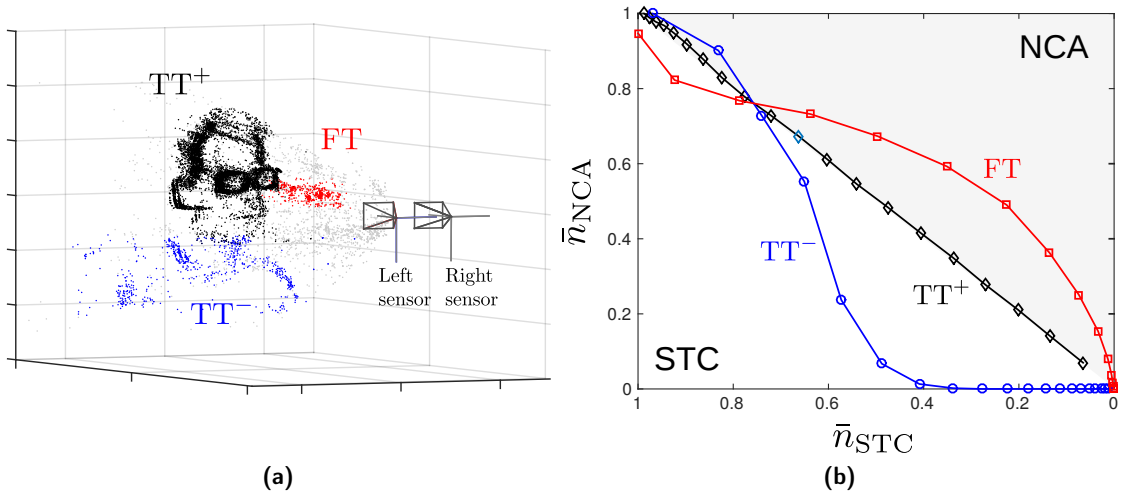


Figure 5.11: Behavior of type III disparity detectors in the presence of false targets. **(a)** Reconstructed scene showing matched disparity events and the ROIs: fast true (TT^+), slow true (TT^-) and false targets. **(b)** Normalized ROI sizes for the two methods.

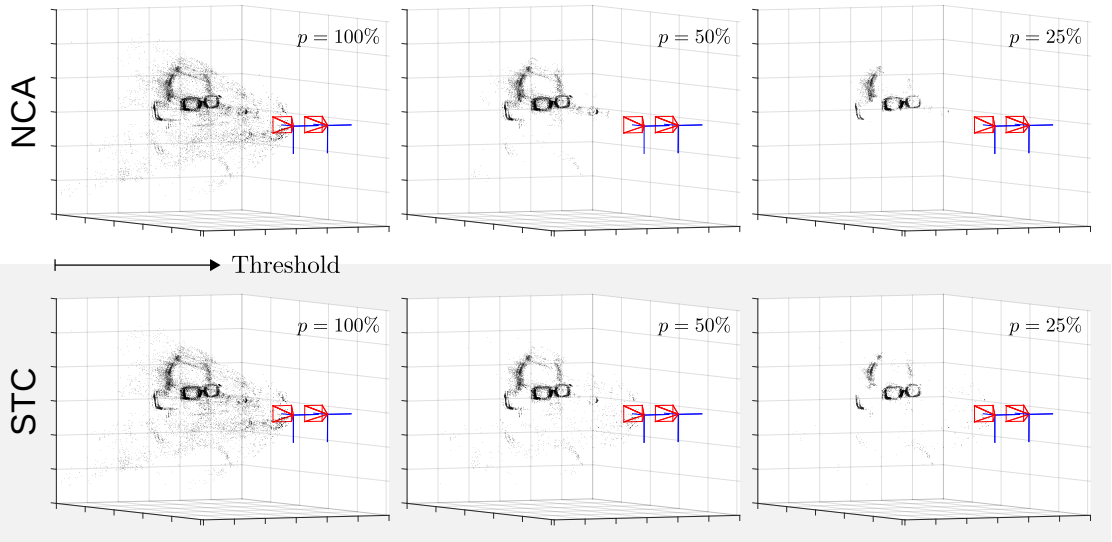


Figure 5.12: Qualitative comparison of the NCA and STC method. Disparity events generated with both methods are shown for different acceptance rates (threshold) accordingly. The STC algorithm shows better selectivity for slow targets (e.g. shoulders) while being less responsive to false targets (e.g. ambiguity of glasses).

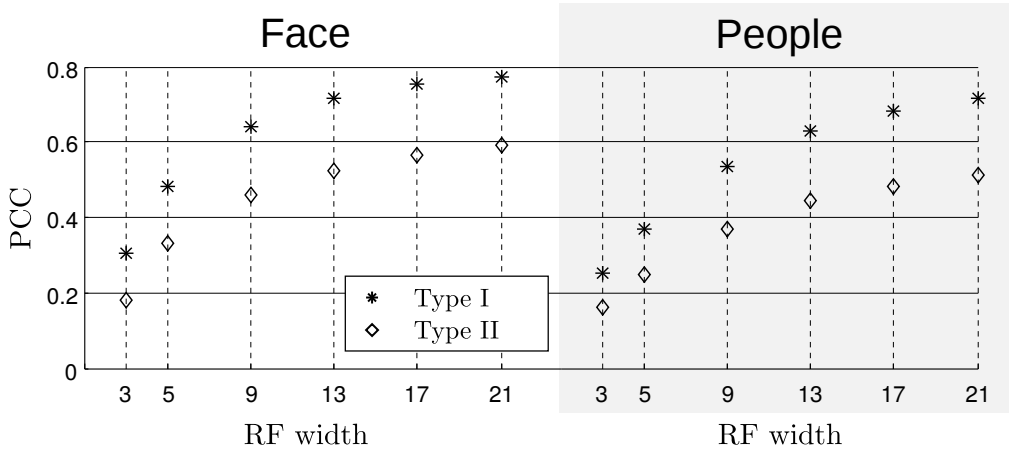


Figure 5.13: Evaluation of neural responses from two types of disparity detectors with (I) and without inhibition (II). The PCC reflects how well the neuron approximates the covariance of temporal images with respect to the width of the receptive field. Evidently, the type I disparity detector produces a closer approximation of the covariance of temporal images.

Type I and II disparity detectors

Type III disparity detectors use the direct inhibitory input from retinal cells to reduce the neuron's response at locations of non-coinciding interocular evidence. Previously, it was observed that activity in the plane of constant horizontal cyclopean position E_x can be used as an alternative cue when the spatiotemporal attributes of the stimuli do not comply. In order to corroborate this hypothesis, two further types of disparity detectors are examined. Type II is the simplest detector that only receives excitatory input from coincidence of constant disparity; thus, it constitutes the baseline behavior. Additionally, type I receives inhibitory input from coincidences within E_x. The experimental procedure was the same as before. This time, however, two DAVIS sensors with lower spatial resolution were used. The experiment was repeated for two different scenes, one of which was similar to the first scene of a moving face, and the other comprising two people passing by each other at different depths. The PCC was used as the statistical measure to determine the degree of correlation between the response of the disparity detectors and the covariance of temporal images. The values of the PCC for both types of detectors are graphed in relation to the receptive field width in Fig. 5.13. For both scenes, the response of type I neurons is more strongly related to the covariance of temporal images. This supports the idea that activity in E_x alludes to counter-evidence to spatiotemporal correlation. It can also be observed that the PCC always increases with the receptive field width. This result is expected, given that when large numbers of coincidence detectors are integrated, the highly non-linear thresholding effect is averaged out.

5.3.2 The resolution of the stereo network to the correspondence problem

The previous examination of different types of disparity detectors revealed that neurons of type I and III compute an approximation of the covariance of temporal images. This makes

such neurons suitable candidates for disparity detectors in the proposed network. While both detectors show similar performance, the type I detector only integrates information from coincidence neurons. As a result, the type I detector is less complex and is thus the preferred choice. Although the detectors themselves show selectivity for true (or similar) targets, they often respond to false targets, particularly when similar features are present. The network attempts to resolve ambiguity through inhibitory interactions among disparity neurons. To assess the overall performance of the stereo network, a dynamic scene was recorded with a pair of DAVIS sensors and the generated events were directly used as spiking inputs to the stereo network. The final output is a combination of the spikes generated from the coincidence and disparity detectors as explained in Section 5.2.6. Note that the terms "spike" and "event" are used interchangeably. Fig. 5.14 shows how the network successfully solves the stereo correspondence problem, even in the case of a complex scene such as two people walking past each other at different depths. The output events of the stereo network $O = \left\{ e_{\mathbf{o}} = (x, y, d) \right\}$ were binned into 30 ms frames with x and y representing the pixel coordinates, and d, the pixel value. The ground-truth data was then completely quantitatively evaluated, as described in section 4.3.3. The quantitative results are also shown in Fig. 5.14. It can be observed that the stereo network performs very well, evidenced by the small local average disparity error $\epsilon_d < 1$ pixels throughout the entire duration of the scene. The disparity error remains largely constant, even at the point when the two people cross each other. At this point, the scene is dominated by large disparity gradients, which is a typical scenario where classical cooperative networks fail. Conversely, the network presented here can resolve these disambiguities by exploiting motion cues. The consistently low disparity error also suggests that the network is fairly robust. Over the entire scene, a total amount of 765,575 3D events were recorded. Using the performance metric proposed for this evaluation method, our network reached a PCM (percentage of correct matches) of 96%. A thorough analysis of different scenes is provided in Section 5.3.6 of this chapter.

5.3.3 Inhibition of stereo ambiguity

In order to identify and suppress false targets, the stereo network relies on the mutual inhibition among disparity detectors. Neurons that receive strong excitatory input, which encode true targets, inhibit other neurons that represent locations on the same line of sight, while remaining compliant with the rule of uniqueness (Marr and Poggio, 1976). Here, this mechanism is referred to as "inhibition of stereo ambiguity" and it is a crucial prerequisite for successfully resolving the stereo correspondence problem. The impact of mutual inhibition proves to be advantageous when the activity of coincidence and disparity detectors are compared. Recall that the spikes of coincidence detectors encode all potential targets, whereas the disparity detectors ideally only represent true disparities. Fig. 5.15 shows a comparison of the activity of coincidence and disparity neurons for a scene in which two subjects, A and B, move at constant depths. The spikes of coincidence detectors are spread over the entire range of disparities, as would be expected when events are falsely matched. Conversely, however, the disparity detectors only show activity in two narrow regions corresponding to the depths of

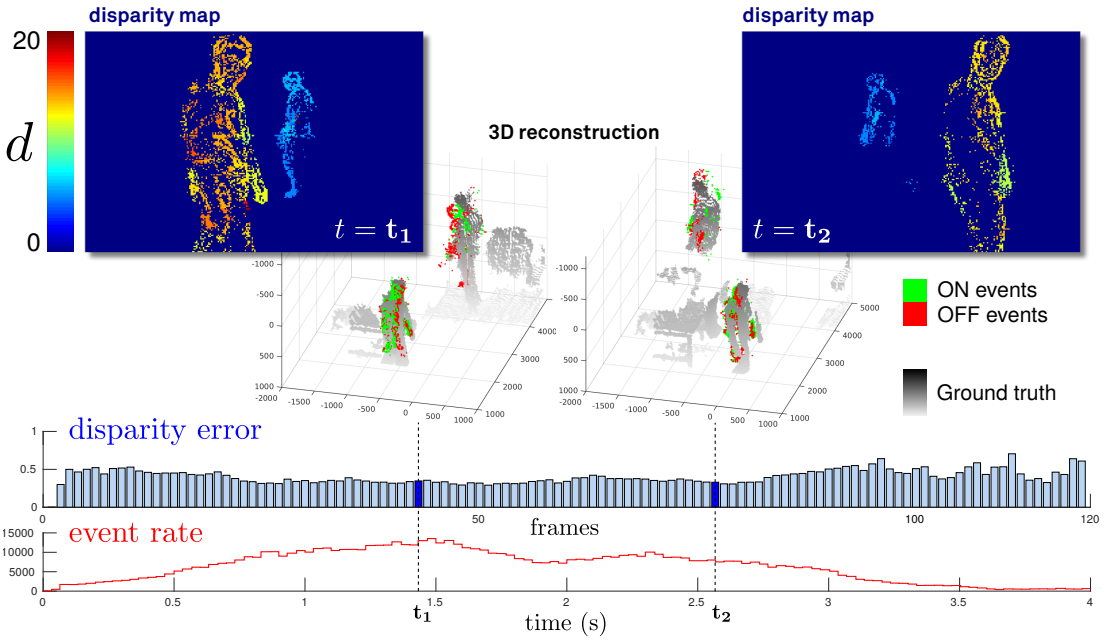


Figure 5.14: Successful resolution of the stereo correspondence problem by the spiking neural network. The recorded scene consists of two people that move in opposite directions at different depths. Here, the two depth maps were generated by binning the output spikes of the network into 30 ms bins at times t_1 and t_2 respectively. The corresponding 3D reconstructions (red and green dots) are overlayed with the ground-truth data obtained from a Kinect sensor (gray). The color encodes the polarity, which is obtained from the event-based sensor.

motion of the subjects A and B, whereas all other disparities (which correspond to false targets) are perfectly suppressed.

5.3.4 Natural disparity tuning curves

It is interesting to take a closer look at the behavior of disparity detectors when mutual inhibition is deactivated. This reveals the tuning functions and selectivity to false disparities of these detectors. A general problem in neuroscience that is associated with measuring tuning functions is that they depend on many aspects of the stimuli. In the past, for example, simple bars or sinusoidal gratings were often used to study the response of biological disparity neurons. In that case, obviously, the tuning functions of the disparity neurons show an unambiguous peak at their preferred disparity. From such an ideal tuning function it is often not clear how the cell responds to the more complex stimuli which occur in natural scenes. In an analogous manner, the tuning curves of the disparity neurons modeled in this thesis would correlate with the structure of the receptive field when tested with simple stimuli. In the case of type I neurons, therefore, broad tuning to horizontal cyclopean position (position invariance) and fine tuning to disparity would be expected. This corresponds to the ideal definition of a disparity detector. The following experiment examines the disparity tuning curves of type I neurons stimulated by natural inputs. When a true disparity occurs, the responses of neurons not tuned to the true disparity are recorded. More specifically, if

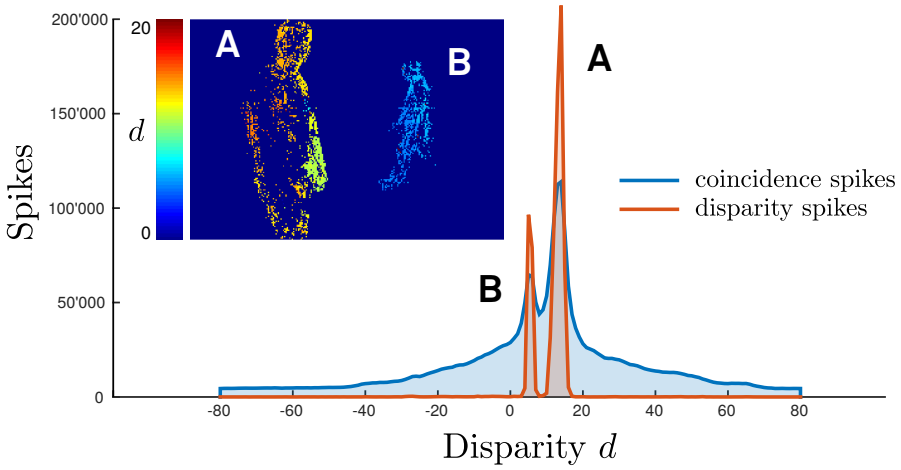


Figure 5.15: Inhibition of ambiguity in the stereo network. Spiking activity of coincidence (brown) and disparity (blue) detectors at varying disparities accumulated over the full duration of the walking scene. The inset shows a disparity map generated from a short section of the scene. The two people are labeled A and B accordingly.

a disparity event occurs at (x, y, d), the neural responses of interest are those which have preferred spatial locations distributed along the directions of x and d. The scene consisted of a face performing all kinds of movement, including translation and rotation. For each of the 600,000 true disparity events, the membrane potentials of 60 neurons with the same cyclopean position but different disparity $d \in [d-30, d+30]$, and 60 neurons with the same disparity but different position $x \in [x-30; x+30]$ were sampled. Not just mutual inhibition, but also the threshold non-linearity of the disparity neurons was removed, in order to study the pure effect of coinciding evidence integration. Finally, all samples were shifted according to the neuron's spatial location in order to receive relative measurements that could be averaged and normalized. The natural tuning curves which were obtained are shown in Fig. 5.16b for varying relative cyclopean positions and in Fig. 5.16c for varying relative disparities. Both tuning curves show the highest response at the true disparity ($\bar{x} = 0, \bar{d} = 0$), while being broadly tuned to $\bar{x}$ and finely tuned to $\bar{d}$. These results indicate that the neurons are very well suited to be natural disparity detectors, as the response does not vary according to the position and shape of the stimulus. Interestingly, the tuning curve in Fig. 5.16c exhibits troughs on either side of the preferred disparity. This is an important characteristic of disparity detectors that suggests that they are not selective to false disparities within their receptive field. Similar characteristics can be observed in the tuning functions of real disparity-tuned cells. As a comparison, the predicted tuning curve from the disparity energy model of Ohzawa (1998)is shown, which also has broad position and narrow disparity selectivity.

5.3.5 Dynamic random dot stereograms

Dynamic random dot stereograms (dRDS) are commonly used in physiological experiments to study the response of disparity-tuned cells in the visual cortex, but are prevalently used

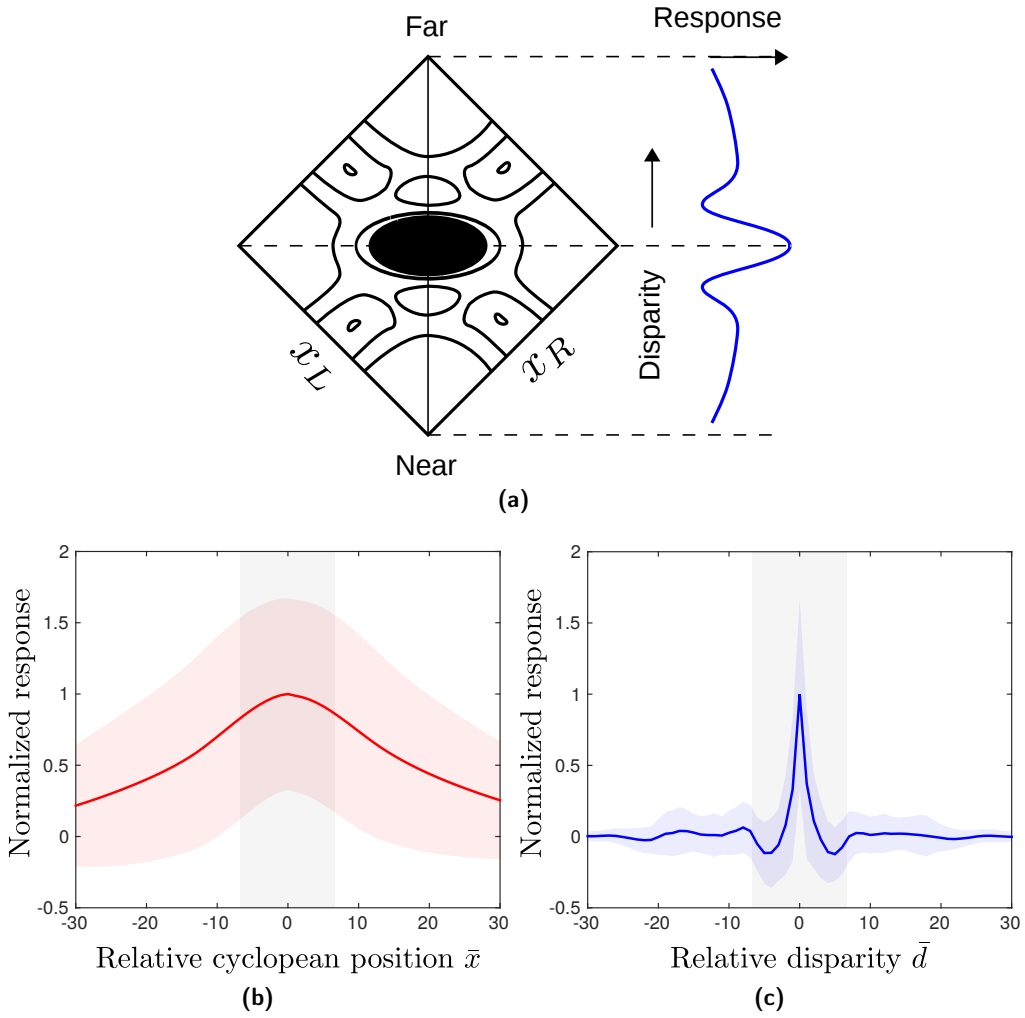


Figure 5.16: Tuning curves of natural disparity detectors. **(a)** Predicted tuning curve from the disparity energy model (adapted from Ohzawa (1998)). **(b)** Average neural response of type I disparity detector neurons for varying relative cyclopean position $\bar{x}$. **(c)** Average neural response for varying relative disparity $\bar{d}$. The solid line represents the mean, while the standard deviation is depicted by the shaded and colored area. The gray-shaded region indicates the width of the receptive field (here $\omega = 13$).

in the field of psychophysics to measure stereo acuity, to give just one example. For the sake of completeness, the proposed network was also tested with a dRDS. The dRDS is generated from a sequence of RDS, updated at 100 Hz. An initial RDS image was computed based on a disparity image of a wireframe cube (Fig. 5.17b) and a random noise pattern, both of which had equal dimensions of 250×250 pixels. Regions containing the same disparities in the disparity image were shifted in the random noise pattern accordingly. In the case of images containing varying disparities, this procedure inevitably leads to areas with undefined disparities, which are observable in the form of shadows in Fig. 5.17c. Subsequent RDS images were generated from the previous one in such a way that there was a 20% chance that each pixel would change polarity. Examples of three subsequent RDS are shown in Fig. 5.17a. The

complexity of the matching problem depends on the frequency at which the RDS images are updated and the probability of a pixel changing polarity. If only a few pixels change their polarity in each consecutive image of a sequence, this could result in a trivial matching problem. This is true even if the update rate is high, assuming that coincidence detectors are tuned for very short temporal delays (such that they only respond to coincidence within a single RDS and not in-between consecutive RDS images). Here, the matching problem was guaranteed to be complex given the parameters and could not be solved trivially from the temporal information provided by the stimulus. This can be easily concluded from the following consideration; as correspondences are only possible on lines of equal y coordinates (epipolar lines), the average number of potential matches between two subsequent RDS images is $0.2 \cdot 250/2 = 25$ (the division by two relates to the fact that there are individual coincidence detectors for each polarity). This represents a considerable degree of ambiguity and thus, is a non-trivial matching problem. The final response of the network is illustrated in the form of an accumulated disparity map in Fig. 5.17c. Based on a qualitative comparison between the disparity map and the ground-truth data, it can be qualitatively observed that the proposed network solves the stereo correspondence problem, even for a relatively complex dRDS containing highly varying disparity gradients (as is the case for the wireframe cube).

5.3.6 Stereo matching performance

In the subsection which follows, the matching performance of the stereo network is assessed for three different dynamic scenes involving a moving face (1), two people moving in opposite directions (2), and a person performing martial arts (3). The results are qualitatively shown in Fig. 5.19, Fig. 5.20 and Fig. 5.21 respectively. The disparity error was computed as explained in Section 4.3.3 and the histograms are individually shown in Fig. 5.18 for all scenes. The distribution of disparity errors fits very well a half-normal distribution:

$$f(x) = \frac{\sqrt{2}}{\sigma_d \sqrt{\pi}} \exp\left(-\frac{x^2}{2\sigma_d^2}\right) \qquad x > 0 \tag{5.22}$$

with a mean of $\mu_d = \sigma_d \sqrt{2}/\sqrt{\pi}$. Generally, it can be observed that the stereo network performs equally well for all scenes. Scene (2) yields the best results, while scenes (1) and (3) yield a slightly higher percentage of incorrect matches. This is likely to be due to the fact that scenes (1) and (3) both contain motion that is not fronto-parallel, whereas in scene (2), the two people are walking perfectly parallel to the baseline. The relatively constant and low disparity error throughout all the scenes suggests that the matching performance is very robust and largely independent of how fast the objects move (illustrated by the event rate). In order to quantitatively measure performance, the percentage of true matches is computed, whereby a match is considered true if the associated 3D event had a disparity error of less than one pixel. This corresponds to the standard PCM of $\delta = 1.0$. Furthermore, the PCMs of $\delta = 2.0$ and $\delta = 3.0$ have been determined. These measures were found to be very high, indicating that although

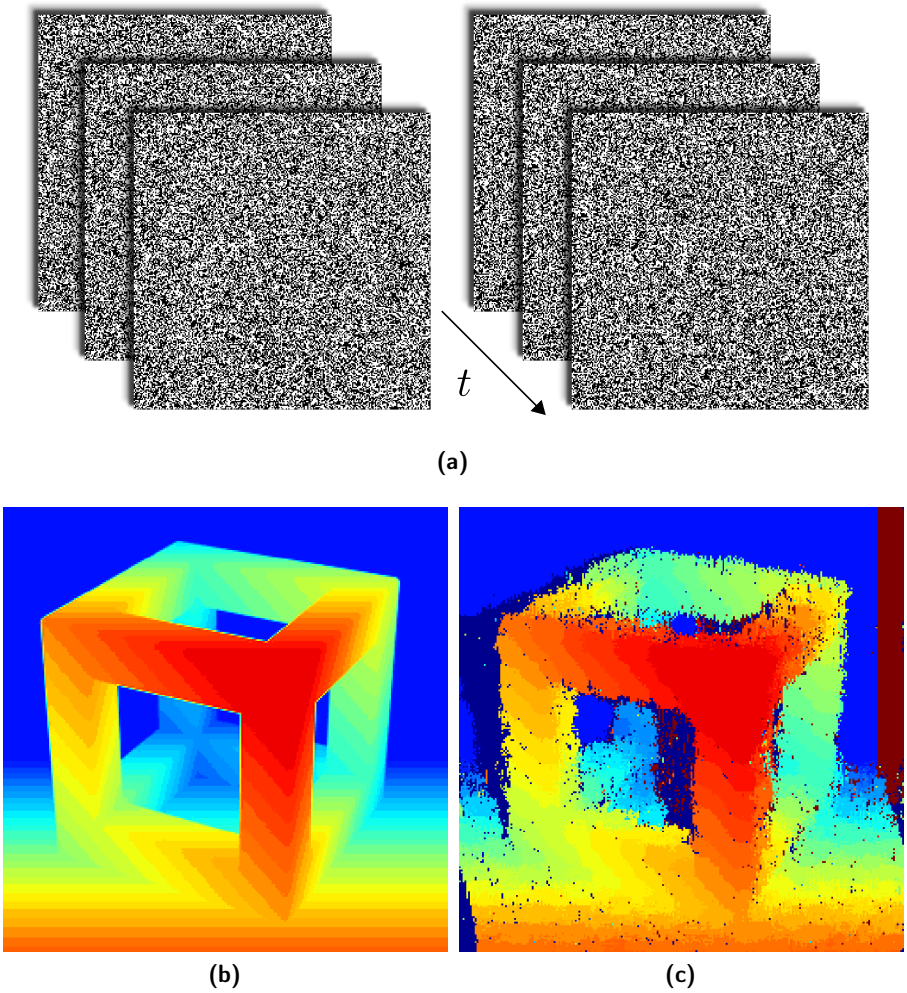


Figure 5.17: The stereo network's response to a dynamic random dot stereogram (dRDS). **(a)** Schematic of the dRDS stimulus for the left and right eye. **(b)** Ground-truth disparity image. Disparity is encoded by color ranging from near (red) to far (blue). **(c)** Disparity map generated from accumulated responses of the network while the dRDS stimulus was presented for 1 second.

the majority of falsely matched targets are inaccurate, they are roughly correct. A summary of the results is listed in Tab. 5.1. Tab. 5.2 lists the scene statistics and the averaged results. All scenes contained a more or less equivalent number of input events, n_l and n_r, from the left and right sensors respectively. The number of coincidences n_c is similar for all scenes, which suggests that the correspondence problem is equally complex. Finally, the number of disparity events n_d matches the number of input events very well, which is desirable (unity gain). The average disparity error for the scene is denoted by $\bar{\epsilon}_d$ and the depth error by $\bar{\epsilon}_Z$ accordingly. It is clearly visible that $\bar{\epsilon}_Z$ depends on the proximity of the scene objects. In scene (2), for example, the second person was located very far away (>4m) from the cameras. Conversely, the average disparity error $\bar{\epsilon}_d$ is independent of proximity and remains constant for all scenes.

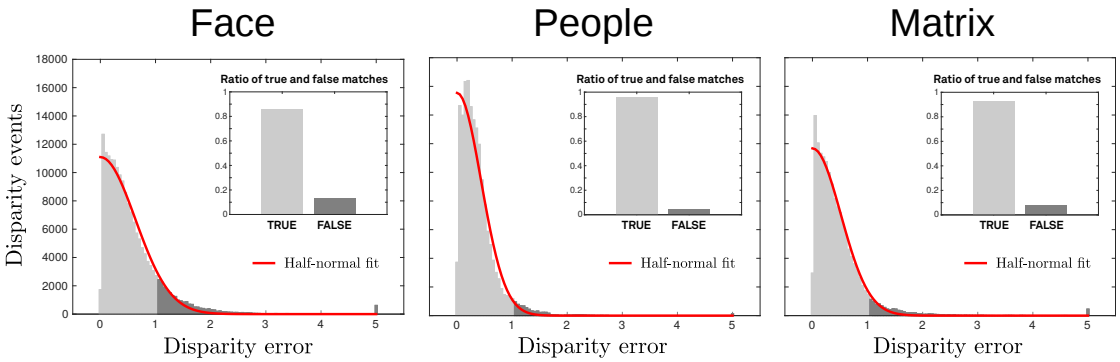


Figure 5.18: Disparity error histograms for all three scenes. Events corresponding to true matches (disparity error <1 pixel) are shaded light gray. All disparity errors that are greater than 5 pixels are contained in the last bin. A half-normal distribution was fitted to each of the histograms (red curve). The insets show the ratio of true and false matches. The "people" scene shows the best observable performance, evidenced by the ratio of true and false matches and the narrowness of the distribution of disparity errors.

Table 5.1: Summary of results.

Scene	PCM ($\delta = 1.0$)	PCM ($\delta = 2.0$)	PCM ($\delta = 3.0$)
Face	86.3%	97.7%	99.2%
People	95.9%	99.6%	99.9%
Matrix	92.3%	98.7%	99.5%
Average	**91.5%**	**98.7%**	**99.5%**

Table 5.2: Scene statistics and experimental results.

	Scene statistics					Averaged results				Fit	
Scene	T	n_l	n_r	n_c	n_d	$\bar{\epsilon}_d$	s.d.	$\bar{\epsilon}_Z$	s.d.	μ_d	σ_d
(1) Face	4.5	685k	518k	3'538k	699k	**0.64**	**2.04**	18.82	24.13	**0.50**	**0.63**
(2) People	4	666k	611k	3'250k	766k	**0.37**	**0.42**	85.77	110.08	**0.35**	**0.44**
(3) Matrix	3.0	557k	430k	3'174k	582k	**0.53**	**1.91**	32.57	32.38	**0.41**	**0.51**

5.3.7 The effect of precise temporal dynamics

The final experiment examines the effect of the temporal precision of spike timing on stereo matching performance. For this purpose, a dynamic scene was chosen that comprised two vertical bars, moving in opposite directions on fronto-parallel planes separated in depth. This is a typical scenario where a static snapshot of the scene would produce identical features at the individual positions of the stimuli. Thus, the ambiguity cannot be resolved without considering motion. The scene was recorded with a pair of DVS sensors which have a temporal resolution well below 1 ms. The retinal events were then temporally binned and entire packets of events (within the same time bin) were fed into the stereo network simultaneously. Thus, the temporal resolution δt of the input was directly determined by the size of the temporal binning window. The experiment was repeated for varying δt and the network's response was recorded.

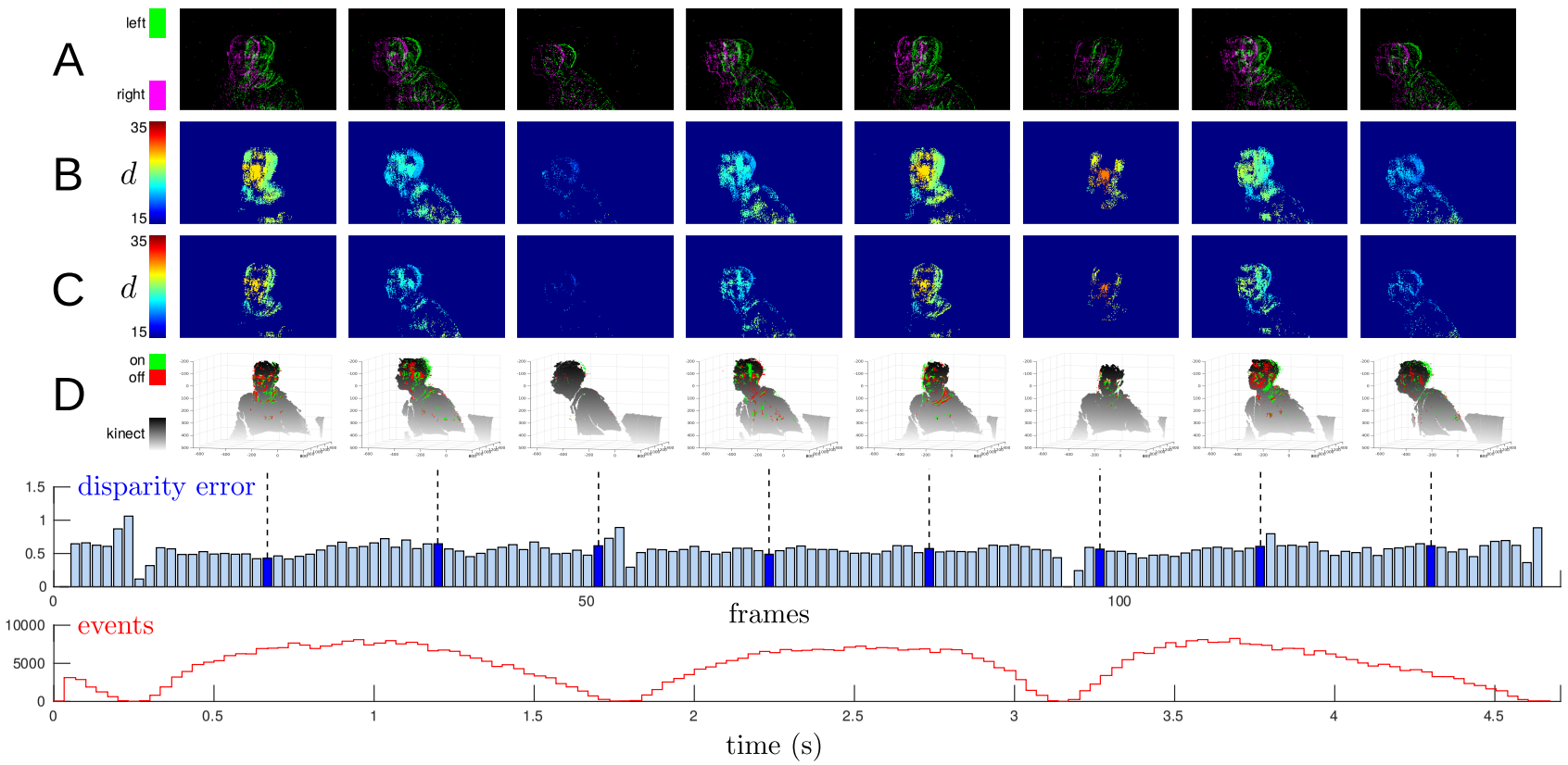


Figure 5.19: Qualitative and quantitative results of scene (1). This scene comprises a face that first rotates to the right (from its point of view) followed by a rotation to the left while moving towards the cameras. Finally, it rotates again rightwards and recedes. The scene lasts 5 seconds. On the bottom, the disparity error and input event rate are plotted over time. The disparity error remains largely constant even when the input event rate strongly varies. The series of images show the input (A), the network activity (B,C) and 3D events (D) at the frames/time indicated. (A) Combined frames of accumulated input from the left (green) and right (purple) sensors. (B) Disparity maps generated from accumulated unfiltered disparity events. (C) Disparity maps generated from accumulated filtered disparity events. As explained in the text, filtered disparity events occur immediately after a coincidence event with the same disparity space coordinates. These events are considered to be the final output of the network. (D) 3D events generated by triangulating disparity events, whose polarity is color-coded in green (ON) and red (OFF). The ground-truth point cloud which was obtained from the Kinect is also shown (gray).

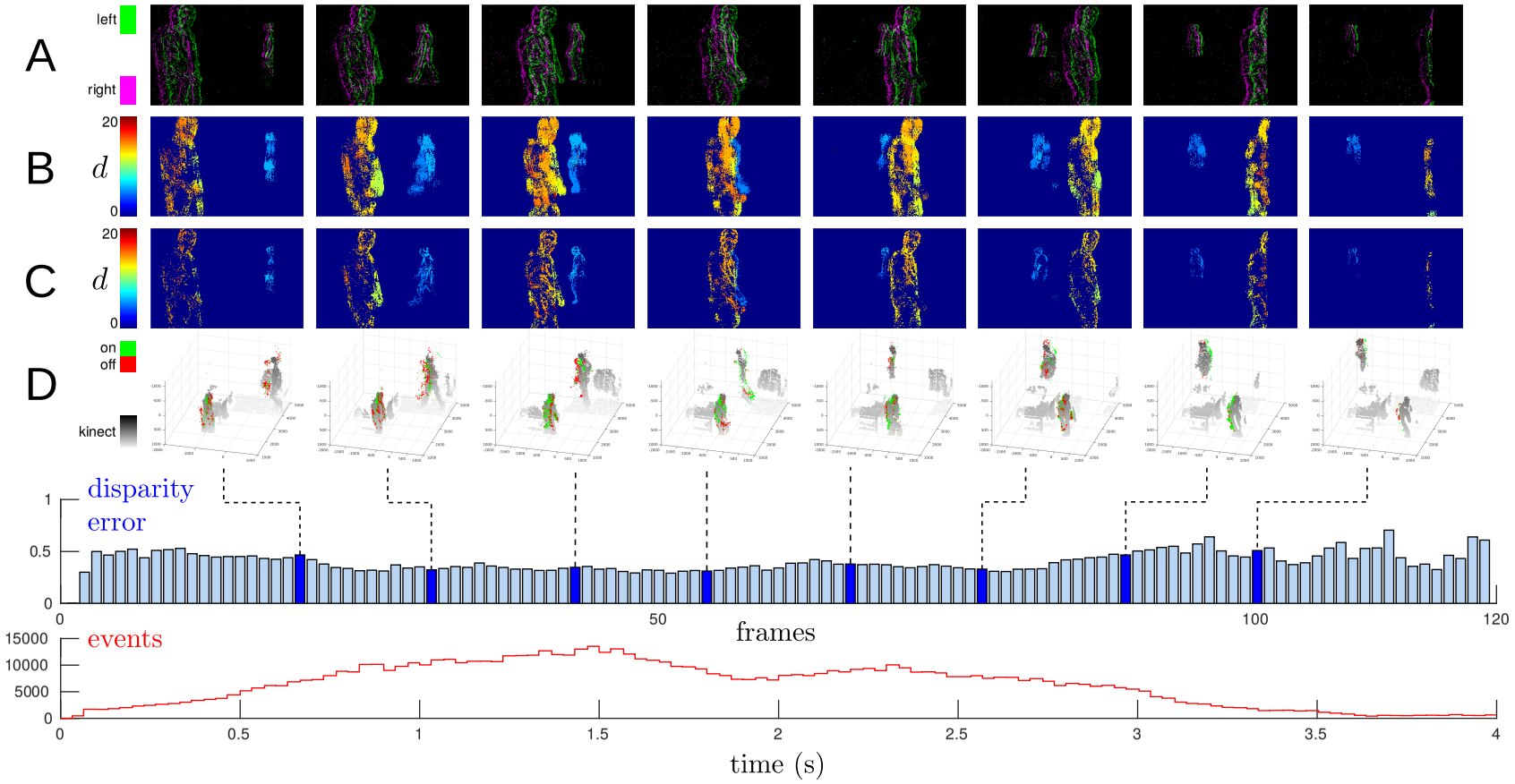


Figure 5.20: Qualitative and quantitative results of scene (2). This scene comprises two people that are walking in opposite directions at different depths. The scene lasts 4 seconds. At the bottom, the disparity error and input event rate are plotted over time. The scene features large disparity gradients (at the point when the people cross each other) and is therefore considered a difficult matching problem. As can be seen from the disparity maps located in the middle, the network unambiguously detects both people. This is explained by the strong motion cues that are present in this particular scenario. The series of images show the input (A), the network activity (B,C) and 3D events (D) at the frames/time indicated. (A) Combined frames of accumulated input from the left (green) and right (purple) sensors. (B) Disparity maps generated from accumulated unfiltered disparity events. (C) Disparity maps generated from accumulated filtered disparity events. As explained in the text, filtered disparity events occur immediately after a coincidence event with the same disparity space coordinates. These events are considered to be the final output of the network. (D) 3D events generated by triangulating disparity events, whose polarity is color-coded in green (ON) and red (OFF). The ground-truth point cloud which was obtained from the Kinect is also shown (gray).

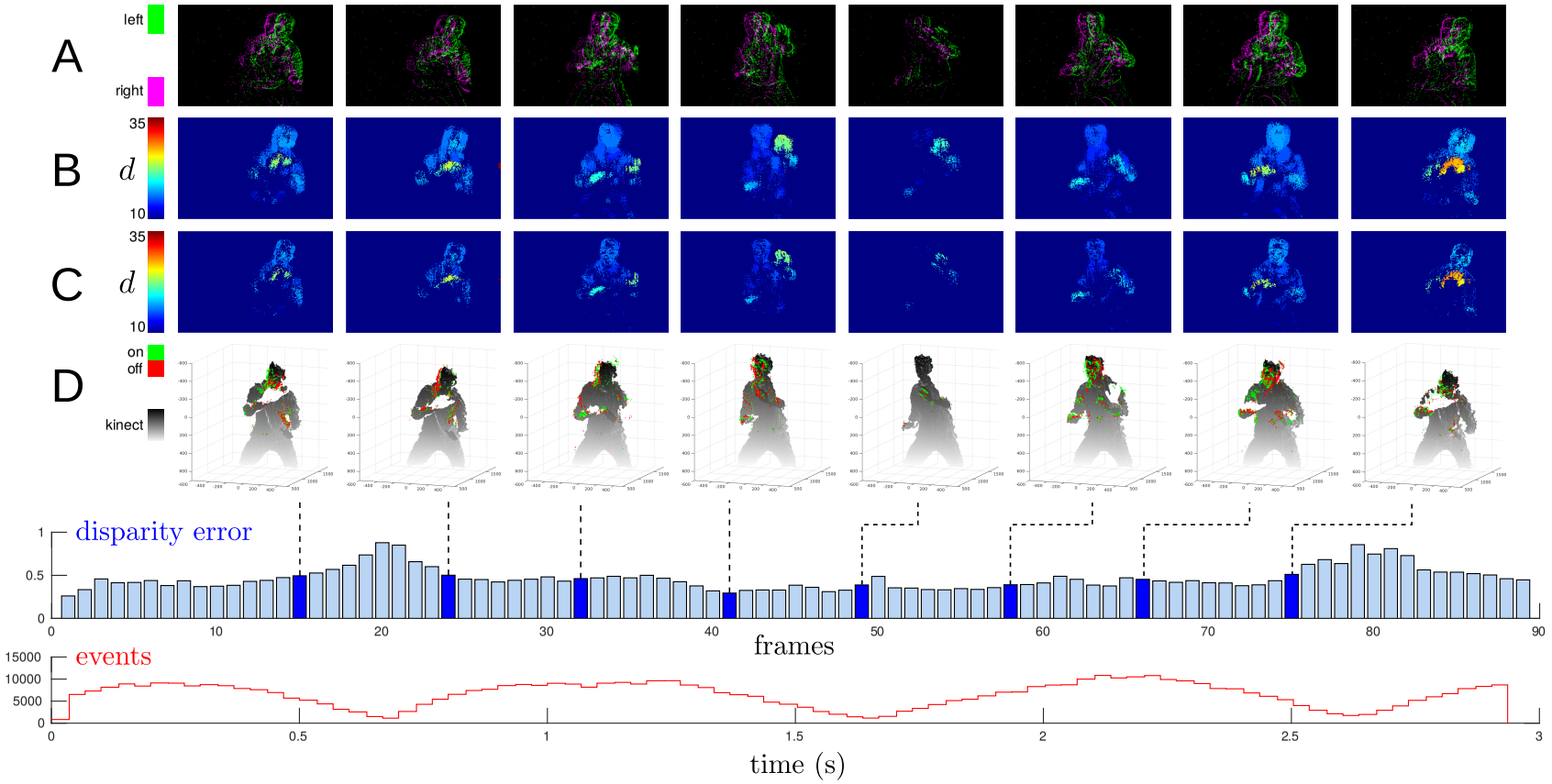


Figure 5.21: Qualitative and quantitative results of scene (3). This scene comprises a person performing martial arts. The scene lasts 3 seconds. At the bottom, the disparity error and input event rate are plotted over time. This scene combines large disparity gradients and various motion cues. Although the hands are partially moving forwards and backwards, the network is still capable of solving the correspondence problem. Here, it can be observed that the disparity error slightly increases when the person moves more slowly (decreased event rate). This is because there are no motion cues at these points, which are required to correctly match large disparity gradients. The series of images show the input (A), the network activity (B,C) and 3D events (D) at the frames/time indicated. (A) Combined frames of accumulated input from the left (green) and right (purple) sensors. (B) Disparity maps generated from accumulated unfiltered disparity events. (C) Disparity maps generated from accumulated filtered disparity events. As explained in the text, filtered disparity events occur immediately after a coincidence event with the same disparity space coordinates. These events are considered to be the final output of the network. (D) 3D events generated by triangulating disparity events, whose polarity is color-coded in green (ON) and red (OFF). The ground-truth point cloud which was obtained from the Kinect is also shown (gray).

The PCM could be easily determined since the stimuli moved at known fixed depths. Fig. 5.22b illustrates how the network performed depending on the degree of temporal precision. At first, the PCM is nearly perfect when the temporal precision is high (small δt), but it drastically decreases to chance level when the temporal precision is lowered (increasing δt). Fig. 5.22a shows two disparity histograms for the the minimal ($\delta t = 1$ ms) and maximal ($\delta t = 200$ ms) temporal resolutions that were considered in this experiment. In the minimum case, only disparities at the two positions corresponding to the depths of the stimuli are observed. In the histogram, false disparities are prevalent, which are distributed throughout the entire range of disparity space. For this particular experiment, the *critical zone* in which performance drops is from 20 ms to 80 ms. In general, such a critical zone exists for every scenario, although it obviously depends on the stimuli and the network parameters. In fact, within the critical zone, coincidences corresponding to false disparities caused by temporal imprecision become increasingly prevalent. Such false coincidences do not fall into the excitatory zones of the disparity detectors with the preferred disparity at the positions of true targets. When temporal precision is even lower, the disparity neurons integrate spatiotemporal evidence in a way which is increasingly biased by more recent coincidences. This eventually leads to the point where their response is solely determined by spatial correlation without considering motion cues. These observations suggest that precise temporal dynamics are beneficial, particularly in the presence of fast moving objects. However, when temporal precision is increased enough to discern between time differences that are smaller than the interocular delay of the coincidence detectors, this does not add further benefit.

5.4 Discussion

This chapter proposed a spiking neural network to solve the stereo correspondence problem, which relies on event-based data from neuromorphic vision sensors. In the section which follows, the proposed network is discussed in relation to accepted models and concepts from neuroscience. It is important, however, to understand that the proposed model is an abstract simplification of its biological counterpart. One fundamental difference is that the proposed network operates using precisely-timed, temporal contrast events from neuromorphic vision sensors. Thus, the network computes *transient* rather than sustained responses, which are similar to those which occur in a certain type of retinal ganglion cell (Y-ganglion cell). The transient responses are represented by single events; thus, the timing of these events is important and plays a crucial role in the stereo correspondence process. In contrast, the vast majority of models of stereopsis are based on mean firing rates, whereby precise spike timing is not important. In this context, the behavior of V1 neurons are very well described by tuning functions that predict the neuron's firing rate in response to a given stimulus in its receptive field. Such tuning functions were found to be well predicted by Gabor filters, which can explain characteristics of V1 cells such as orientation and spatial frequency tuning. Accordingly, stereopsis models are based on binocular energy neurons that combine monocular Gabor filters and can predict the responses of disparity-tuned binocular cells in V1 very well. The

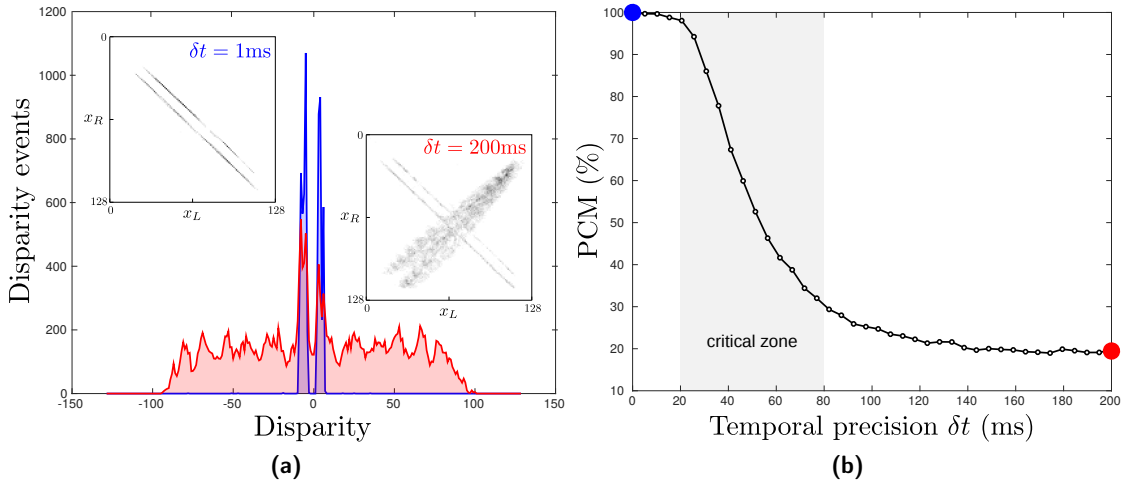


Figure 5.22: The effect of precise temporal dynamics on stereo matching performance. **(a)** Two histograms of disparity events from the stereo network for a scene comprising two moving bars at constant depths. The histograms are shown for scenarios with high (blue) and low (red) temporal resolution. The insets show the distribution of disparities in disparity space (viewed from above). In the first scenario, the trajectories of the two moving stimuli are correctly reconstructed, whereas false matches are prevalent in the second scenario. **(b)** Matching performance of the network when temporal resolution is altered. The two scenarios from (a) are highlighted accordingly.

significant mechanism of phase and position disparity are direct consequences of the way in which receptive fields and tuning functions are described in neuroscience. While these models definitely explain many characteristics and aspects of the physiology of stereopsis, they do not account for important mechanisms of neural systems such as *temporal dynamics* and *spiking neurons*. It could be argued that the model presented in this study is abstract and biologically implausible because it incorporates neither orientation and frequency tuning, nor phase and position disparity mechanisms. These characteristics are based on the perception of *spatial contrast*, whereas the proposed model solely responds to *temporal contrast*. Instead, the proposed model assigns an important role to the temporal dynamics and transient responses possibly involved in stereopsis and makes explicit use of spiking neurons. Different interpretations of the model are possible based on architectural and functional considerations. For example, it could be functionally combined with the energy model. In this manner, higher order disparity detectors could integrate transient responses of energy neurons in the way outlined in the proposed model. In this case, an event as defined in the proposed framework would not directly represent a neural spike. Instead, it would represent a kernel indicating a transient response. On the other hand, events could indeed represent single spikes that directly signal changes in temporal contrast, as exemplified by Y-ganglion cells. If this were the case, the proposed model could directly describe a simple neural network for coarse and fast stereopsis, which could coexist in parallel with a more precise and substantial stereopsis process (as described by the disparity energy model). Such a simple stereo process could be

involved in vergence eye movement as part of the magnocellular system. Taking these interpretations into consideration, the following subsections highlight commonalities between the proposed model and the physiology of stereopsis and attempt to lay the foundation for a unified model of stereopsis and machine vision.

5.4.1 Stereo from correlation and the integration of motion cues

Many stereo vision algorithms are based on a local correlation measure between image features, such as the sum-of-squared-differences (SSD), for example. Correlation models of stereo correspondence are also prevalent in neuroscience. Although Panum's fusional area and the disparity gradient limit (Burt and Julesz, 1980) were the first indications that stereopsis is subject to a correlation mechanism, modern literature also supports this idea (Nienborg et al., 2004; Banks et al., 2005). Disparity detectors at early stages are believed to be tuned to small patches of uniform fronto-parallel disparities. When combined, they could be used to percieve more complex disparity structures in higher areas (DeAngelis, 2000). This proposition conforms with the first observation made by Marr and Poggio (1976). After describing the prevalence of smoothly distributed disparities in natural scenes, the authors propose a rule of excitation among cells of equal disparities in their cooperative network. In the method described here, this mechanism was adapted and supplemented with an inhibiting mechanism, showing that the proposed neurons performed an approximation of local covariance of spatiotemporal image features. Spatiotemporal information strengthens the correlation even in the case of non-fronto-parallel disparities (when the motion is fronto-parallel). In other words, the stereo network naturally exploits motion cues to overcome the limits of stereo matching based on spatial correlation alone. This is achieved simply by using a different concept to encode visual information. In neuroscience, it is well known that motion cues can play a crucial role in solving the correspondence problem. However, it is not clear how and where they are integrated in the brain. Experimental studies in macaque monkeys and humans suggest that the primate brain needs to integrate several cues ranging from low to high cortical levels in order to solve the correspondence problem (Janssen et al., 2003; Preston et al., 2008). Among these cues, motion (that is integrated in mid-level visual areas) is one of the most important. The hybrid energy model (Qian and Zhu, 1997; Anzai et al., 2001) demonstrates that motion and stereo are jointly encoded at an early cortical stage and emphasises the importance of the relationship between motion and depth. Despite this, however, there are surprisingly few studies that examine how motion benefits the correspondence problem (a notable one is Bradshaw and Cumming (1997)). In this area of research, the computation of motion in depth (stereomotion) has been more intensively studied in the stereopsis literature (Rokers et al., 2009; Cottereau et al., 2014). Two models have been proposed. The changing disparity (CD) model derives stereomotion from the rate of change of disparity from binocular cues. The interocular velocity difference (IOVD) model is based on monocular motion signals with opposing directions (which is the case for an approaching or receding object). It has been shown that the IOVD cue plays a dominant role in perceiving stereomotion (Brooks, 2002). This corresponds with the model proposed here, which predicts that the disparity of approach-

ing and receding objects is poorly perceived because suppressive coincidence neurons are activated in the plane of constant cyclopean position. For this reason, the proposed model supports the idea that motion in depth should be computed from cues other than the rate of change of binocular disparity.

5.4.2 Relation to cortical disparity detectors

Ohzawa et al. (1990) famously first outlined the ideal characteristics for disparity detectors and developed a model which proved to be highly congruent with recordings from cortical cells. The ideal properties of what the authors termed "the archetypal disparity detector" were described as follows: "First, the disparity selectivity must be much finer than that predicted by the size of the receptive fields. Second, the preferred disparity must be constant for all stimulus positions within the receptive field. Third, incorrect contrast polarity combinations should be ineffective if presented at the optimal disparity." (Ohzawa et al., 1990). The resulting disparity energy model describes how complex cells can emerge disparity selectivity by integrating various simple cells. It can be observed that the disparity detectors proposed in the present study fulfill not only the properties of the archetypal detector, but also show strong analogies to the disparity energy model. While in the energy model, complex cells integrate the response of simple cells with constant disparity, the detectors proposed in the present study integrate coincidence neurons with constant disparity. A fundamental debate exists over whether V1 neurons are sophisticated enough to distinguish correct and false matches. The energy model does not account for such a discrimination and various studies with anti-correlated RDS have shown that these type of stimuli also activate disparity selective neurons in V1 in such a way that their tuning curve is inverted (Ohzawa et al., 1990; Livingstone and Tsao, 1999). This complies with the prediction of the energy model and the conclusion was drawn that additional processing beyond the striate cortex is required in order to solve the stereo problem. However, it has been noted that the response to anti-correlated stimuli was often weaker, than for correlated stimuli (Ohzawa et al., 1997; Cumming and DeAngelis, 2001). Recent studies support the idea that V1 neurons play a more important role in discriminating correct and false targets and that the energy model needs to be revised. A computational study proposed that by adding suppressive elements to the energy model, responses to false matches may be attenuated (Read and Cumming, 2007). This has been confirmed with experimental results from the V1 neurons of monkeys (Tanabe et al., 2011). Interestingly, the work in hand draws the same conclusion from an entirely computational approach, without any prior knowledge of the physiological findings. The additional use of inhibiting elements in the proposed disparity neuron model means that in effect, the neurons compute a form of correlation that behaves similarly to the covariance of interocular spatiotemporal images. Thus, the neurons clearly show attenuated responses to false targets. In the model presented here, the elements of inhibition are coincidence neurons from the plane of constant cyclopean position E_x. These coincidence detectors receive input from retinal cells that are positioned relatively similarly in the two retinas to those that provide the input to two biological cells: phase-detectors in anti-phase and position-detectors with different preferred disparity. In stereopsis research,

exactly these type of detectors have been proposed to represent the suppressive mechanism in V1 neurons that helps to solve the correspondence problem (Read and Cumming, 2007; Haefner and Cumming, 2008; Tanabe et al., 2011).

5.4.3 Where is the stereo correspondence solved in the brain?

Although there is a strong belief that the stereo correspondence problem is solved beyond the striate cortex (Cumming and Parker, 1997; Cumming and DeAngelis, 2001; Tanabe et al., 2004, 2005; Janssen et al., 2003), the model proposed here suggests that this could also theoretically happen within the striate cortex through an interaction of disparity-tuned V1 complex cells. It has been shown here that the proposed disparity detectors deal with the correspondence problem by more strongly responding to correct matches, while attenuating the response to false matches. The decisive suppression of false matches, however, is made possible by the mutual inhibition of disparity detectors along the same lines of sight. It has been experimentally demonstrated here that this inhibitory interaction successfully suppresses false disparities and that no further processing is required to unambiguously code true disparities. This suggests that the stereo correspondence problem can be solved by employing three mechanisms: coincidence detection, disparity detection and mutual inhibition. All three of these mechanisms could occur together in V1 in the form of simple cells, complex cells and recurrent connections. Alternatively, the coincidence detection mechanism does not necessarily have to be realized with neurons, but could be directly implemented on dendritic branches of complex cells (Alonso and Martinez, 1998; London and Häusser, 2005). Together, these observations suggest that the stereo correspondence problem could be solved at an early stage of visual processing, possibly in V1. Due to the vast amount of evidence that suggests the significant involvement of extra-striate areas, it is argued here that the stereo problem is solved in both visual pathways independently. Based on this idea, the proposed model would explain the process of fast and coarse stereopsis of the magnocellular system (dorsal stream). Conversely, the slower process of fine stereopsis would be part of the parvocellular system and could involve more visual areas along the ventral pathway.

5.4.4 Models of disparity interactions

Beyond mutual inhibition, other forms of interaction among disparity-tuned cells exist across spatial scale. This idea is widespread in the stereopsis literature. Psychophysical evidence of coarse-to-fine interactions suggest a multi-channel model in which disparities at low spatial frequencies facilitate fusion of disparities at high spatial frequencies (Rohaly and Wilson, 1994). Conversely, however, evidence for fine-to-coarse interactions has also been reported (Smallman, 1995). As previously stated, the model proposed here does not explicitly consider varying spatial scales. However, it was observed that the proposed disparity detectors were much more broadly tuned to cyclopean position than the coincidence detectors. In addition to this observation, it can be argued that the proposed network employs two kinds of neurons with differing spatial scales. Firstly, it employs simple high frequency detectors

that respond equally to true and false matches (i.e. coincidence neurons). Secondly, it employs complex low frequency disparity detectors, which respond more strongly to true targets (i.e. disparity neurons). The final output is obtained by combining both responses (see Section 5.3.2) which supports the idea of a coarse-to-fine interaction. On the other hand, the response of the disparity detectors is achieved by integrating multiple coincidence detectors, which also supports the idea of fine-to-coarse interaction. Interestingly, such fine-to-coarse interactions were already proposed previously in the computational model of Mahowald (Mahowald, 1994a), long before the emergence of physiological evidence.

5.4.5 Testable predictions for psychological stereo illusions

A few psychophysical illusions exist that are related to the process of stereopsis such as the Pulfrich effect and the double-nail illusion. It was considered to be beyond the scope of this study to examine a possible relationship with the Pulfrich effect. Whereas the classical Pulfrich effect is simply explained by the interocular delay of the stimulus, this explanation does not hold in the case of a moving object viewed stroboscopically (Read and Cumming, 2005). The effect is closely linked to the perception of motion and depth. Thus, the model proposed here could possibly cast light on the underlying neural mechanism that explains the illusion, which remains unclear. Conversely, the double-nail illusion is well understood. This illusion occurs when two identical objects (for example two nails) are viewed straight ahead at reading distance at the same position, but are separated in depth by a few centimeters. The two objects are perceived as if aligned side-by-side instead of one being behind the other. It was observed that the perception occurs at the positions of "ghost targets" (falsely matched objects). The simple cooperative network can explain this effect. More generally, any matching mechanism based on correlation explains this effect as such mechanisms always prefer the consistent ordering of objects in both views. In the case of the double-nail experiment, the order is reversed and thus, false matches are preferred. Obviously, the proposed model also predicts the double-nail illusion. However, another very interesting prediction of the model relates to moving objects. If it is assumed the two objects are not static, but move sideways at constant depth in opposite directions, the model would predict that the correspondence problem could be solved correctly due the presence of motion cues. The moving objects would then be perceived at the correct positions, even at disparity gradients that exceed the limit of human stereopsis. As a logical consequence, if the two objects were separated in terms of horizontal position but aligned at the same reading distance and moved alternately back and forth, their ghost targets, which would be moving sideways at different depths, would be perceived. It is notable that although the two scenarios described are physically different, they produce perfectly identical projections on the retinas. Further psychophysical phenomena that could be related to the perception of depth and motion are the Ternus and Flash-lag illusions. It would be interesting, therefore, to study these effects in relation to the proposed model.

5.4.6 Neuromorphic hardware implementation

The approach put forward in this thesis originates from the field of computer science. In this chapter, it has been contrasted to the physiology of stereopsis, and supporting evidence for biological models has been presented. A simple, yet comprehensive bottom-up network model that combines many profound key principles from stereopsis has been proposed and it has been demonstrated that such a model can solve the stereo correspondence problem. However, does this study also have implications for computer science? In the field of stereo vision, the correspondence problem has been solved by a variety of approaches. The challenge comes down to performance in terms of latency, computational cost and power consumption. In order to predict the limits of Moore's law, there has been an increasing interest in alternative computing paradigms beyond the classical von Neumann architecture. Most of them aim to distribute processing units and memory using massively parallel architectures. The development of such brain-like computers — the earlier discussed *neuromorphic hardware* — has recently led to new milestones in computer science (Furber et al., 2014; Benjamin et al., 2014; Merolla et al., 2014; Qiao et al., 2015) and they serve as a perfect substrate for the proposed stereo model. In future, such stereo systems could greatly expand the dimensions of computing by outperforming traditional models in terms of latency and power consumption by multiple orders of magnitude. This is likely to be a fundamental aspect of the mobile devices and robots of tomorrow. Thus, the implementation of the proposed model on neuromorphic hardware is discussed in the next chapter.

6 Neuromorphic Real-time Stereo Vision Systems

Traditional hardware based on the Von Neumann architecture is not suitable for implementing massively parallel neural model. The stereo network presented in the previous chapter is an example of such a network. It comprises many units that individually perform simple computations. Thus, only a hardware substrate that can carry out these computations in a highly parallel manner can fully leverage the advantages of this approach. It has been shown that neuromorphic engineering has produced hardware suitable to efficiently emulate neural networks. In the course of this chapter, a versatile, neuromorphic stereo vision system is explained which involves a custom-built, spiking *neuromorphic processor*. Although it is not optimally suited to the task of stereo vision, this analog processor is at the heart of the system which demonstrates a proof of concept.

6.1 A versatile, neuromorphic multi-chip stereo vision setup

The neuromorphic stereo vision system that has been developed in the course of this thesis is schematically illustrated in Fig. 6.1a. The system is designed to be as versatile and accessible as possible so that it can be implemented in a variety of experiments that could be extensively monitored. The structure of the setup is analogous to the human visual system, comprising sensors (retinas), a relay center (LGN) and a visual processing area (V1). The system consists of four functional parts that involve sensing, mapping, probing and processing and are explained accordingly in the following sections. Figs. 6.1b-6.1d show three configurations of experimental setups using the system, involving a *simulation* (b), a *hybrid* (c) and an *emulation* setup (d) . A photograph of the complete setup is shown in Fig.6.2 with the components labeled accordingly.

6.1.1 Sensing

The sensors of the stereo vision system are two DVS, event-based cameras with 128×128 spatial resolution. They are horizontally separated by a baseline distance of about 85 mm

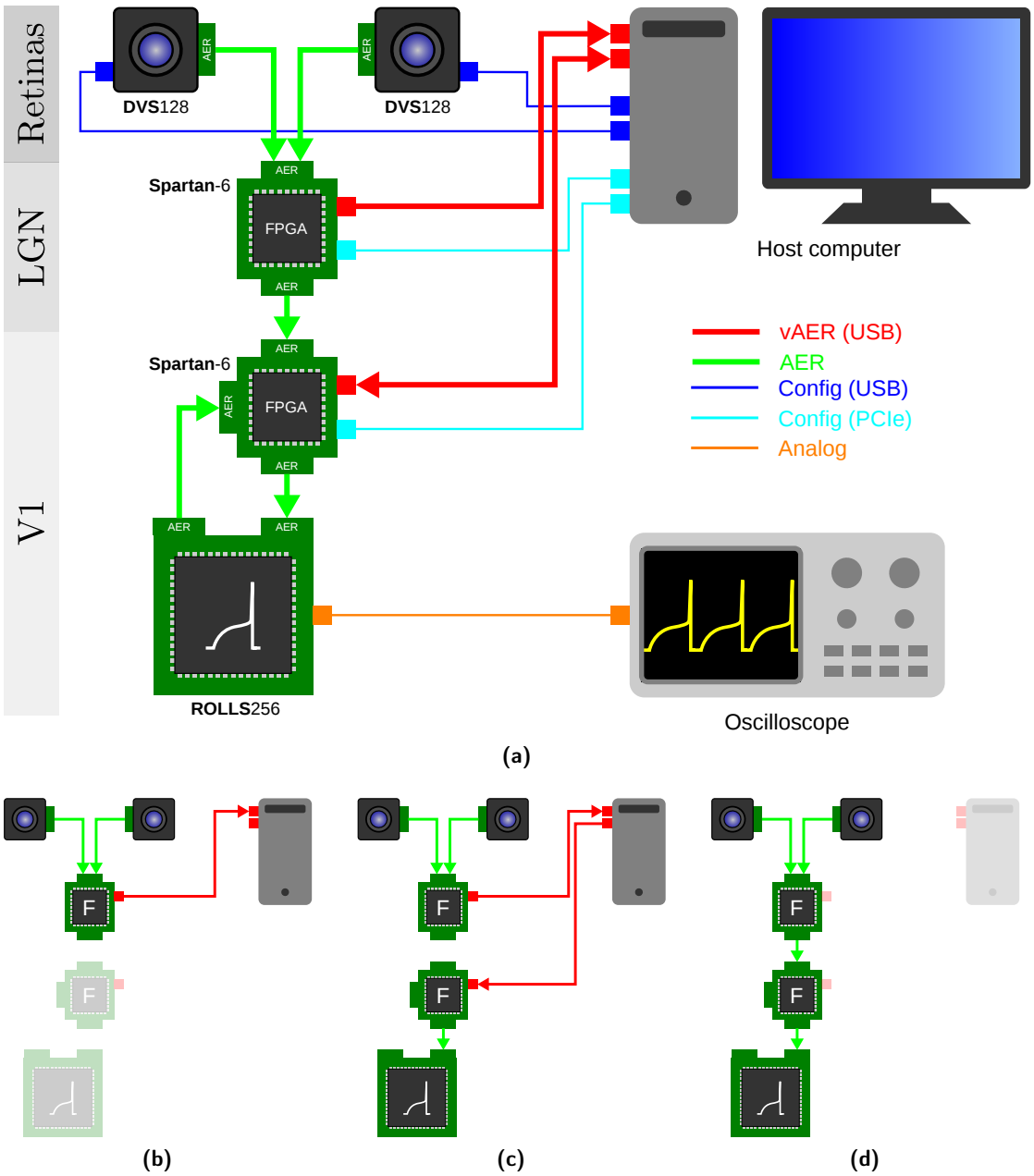


Figure 6.1: Schematic of the neuromorphic, multi-chip stereo vision setup **(a)** The complete setup with all its components. **(b)** Simulation configuration. **(c)** Hybrid configuration. **(d)** Emulation configuration.

which is slightly larger than the pupillary distance of humans (≈ 65 mm). Both sensors are slightly rotated such that they converge on a fixation point which is located about 1 m in front of them. Temporal contrast events from both sensors are directly communicated to the LGN-FPGA board using the AER protocol. The FPGA samples incoming AER requests at a maximum polling frequency of 100 Mhz. A DVS sensor produces event rates of up to 2 Mevent/s, typically

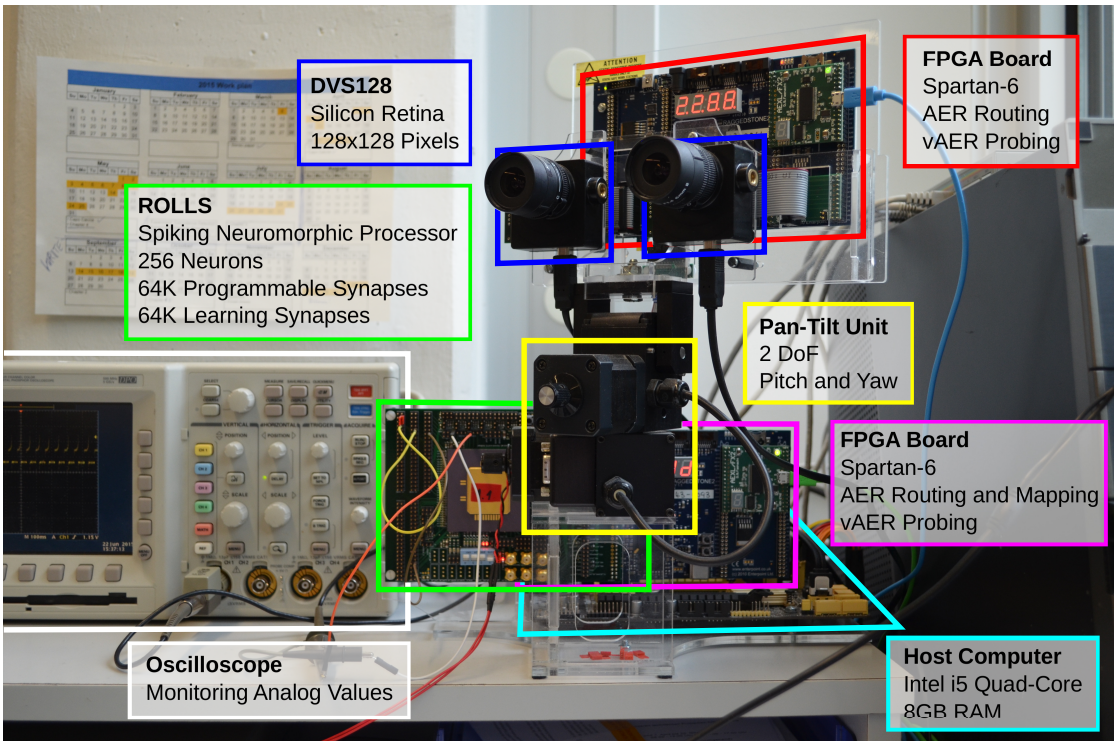


Figure 6.2: Photograph of the neuromorphic multi-chip stereo vision setup.

in scenes with high contrast and only if the sensor itself is moving. Each DVS has an additional programmable logic device (CPLD) and USB microcontroller (Cypress FX2), which allows bias settings to be programmed and provides a direct read-out of events. The user can select which of the two logic devices (the CPLD of the DVS or the common LGN-FPGA) actively controls AER handshaking, while the other one then becomes a passive listening device. Here, handshaking was acknowledged by the LGN-FPGA. As a consequence, the completeness of event-based data is not guaranteed when reading directly from the event-stream on the USB interface of a particular DVS.

6.1.2 Probing

The event streams can be read out at several probing points in the system using the concept of *virtualization*. For this purpose, a logic device samples the events at high frequency and attaches a digital time stamp to each event at the exact time at which it was sampled. These *virtualized events* (vAER) can then be sent over a serial link (using USB) to a host computer where they can either be analyzed, or used as a further input for simulation. If the virtualized events are used for further simulations, it could be argued that such a process does not strictly comply with the concept of real-time computing. The issue is that virtualized events are not guaranteed to be processed at the time they occur. However, in practice such delays are typically very small (<1 ms) and more importantly, the relative timing of events is preserved in the digital time stamps. Furthermore, the term “real-time” in the context of simulation

solely means that the simulation runs at the same speed as a real clock and thus, it functions independently of any possible delays in the input. Nevertheless, in a strict sense there are different meanings to consider and the consequences of real-time processing in neuromorphic systems will be discussed later. The entity that performs the virtualization on the logic device is called the "monitor". The temporal resolution of the monitor is typically set to 1 μs, which is about a tenth of the sensor's temporal resolution. The CPLDs of the individual sensors contain monitors that provide separated virtualized event streams. A merged event stream which combines the output from both sensors is available from the monitor in the LGN-FPGA. Finally, a further monitor in the V1-FPGA provides a virtualized event stream of the output from the spiking neuromorphic processor. The complement of a monitor is called a "sequencer". The sequencer is a digital entity that releases AER events at the precise times predicted by the relative intervals of virtualized events. Virtualized events are sent over a serial link from the host computer to the sequencer. Such a sequencer exists only in the V1-FPGA and it is used to stimulate the neurons of the neuromorphic processor. Lastly, the neuromorphic processor itself has a few analog probes, which can directly measure certain values using an oscilloscope, such as the membrane potential of a neuron or the synaptic current of a synapse . These probes can be digitally controlled so that they connect to a specific neuron or synapse. Probe configuration is handled by the host computer through a PCIe interface.

6.1.3 Mapping

The primary function of the relay center found on the LGN-FPGA board is to merge the two visual event streams and forward them to the V1-FPGA, which is at the origin of the visual processing area. Basic filtering functions can be implemented on the FPGA and controlled via the host computer. The array of possible filter functions include background activity suppression, polarity filtering, selection of the field of view, and stereo rectification. The filtered outputs can be computed within 1-2 clock cycles. Together with the time required to process the input and output, this results in a negligible delay in the event stream ($\delta t \approx 50$ ns) received at the input interface of the visual processing area. The main purpose of the V1-FPGA is to handle and route events between the LGN-FPGA, the neuromorphic processor and the host computer. It contains a mapping entity that consists of a look-up-table and provides the possibility to arbitrarily link sources to targets at the level of a single cell. This allows receptive maps to be simply configured between sensor pixels and spiking processing neurons as they are used in the stereo network. In fact, the neuromorphic processor itself is capable of implementing mapping functions without requiring an external digital entity. However, this functionality is limited to recurrent mapping across neurons. For this reason, an FPGA needs to be used to map the inputs.

6.1.4 Processing

At the heart of the visual processing area is a *re-configurable, on-line learning, spiking* (ROLLS) neuromorphic processor. It contains 256 analog neuron circuits and 128K synapses. The

ROLLS chip processes the AER input strictly in real-time and produces AER output without using virtualization at any point. Typically, V1 processing is completely implemented on the neuromorphic processor. Alternatively, the V1-FPGA can be used to perform an early stage of V1 processing. The exact role of all the hardware devices depends on the configuration of the setup. Generally, three configurations are possible, which are based on two distinct modes of computing.

Simulation

A *simulation* uses a general purpose processor to execute a software model of the neural system. In the context of the stereo vision system, this means that the event streams are virtualized and processed on the host computer. The relevant configuration of the setup is depicted in Fig. 6.1b. The pair of DVS sensors provides independent streams of temporal contrast events, which implements the function of the retinas. These streams are merged on the LGN-FPGA, directly virtualized, and sent to the host computer. From this point onwards, everything ranging from stereo rectification, to the simulation of the behavior of all the components of the stereo network is carried out by software. The simulation is said to run in *real-time* as long as the temporal delay between the real and virtualized time of an output event is consistently small (typically < 30 ms).

Emulation

As an analogue to simulation, *emulation* exclusively uses special purpose electronic circuits to imitate the behavior and mechanisms of the neural system. To achieve this, the setup is configured to use all available hardware devices, as illustrated in Fig. 6.1d. Throughout the entire system, events are directly processed and communicated at the time they occur. This is an important characteristic of *strict* emulation, which implies that no virtualization is used at any point. In the context of emulating the stereo vision system, the individual hardware components carry out the following roles. As before, the DVS sensors provide independent visual event streams. The streams are merged and filtered if necessary. Stereo rectification is performed in the FPGA device representing the LGN. The combined, rectified event stream is then forwarded to the visual processing area (V1). Here, the V1-FPGA device and ROLLS processor can take on different tasks. One possibility is that all processing, from coincidence to disparity detection, is carried out by the neuromorphic processor. In this scenario, the sole purpose of the FPGA device at this stage is to carry out the required mapping function between retinal cells and V1 neurons. Alternatively, the neuromorphic processor could play the role of the complex disparity neurons, while the coincidence detection behavior of the simple cells could be carried out by the FPGA device. In either of these emulation configurations, the host computer is only used for monitoring.

The system is said to be a *hybrid* configuration if the neuromorphic processor is used for emulating the stereo network, but the functions of the FPGAs, such as filtering, rectification,

mapping and partial coincidence detection, are performed by the host computer instead. This configuration is shown in Fig. 6.1c. Again, the merged visual event stream is virtualized at the point it reaches the LGN-FPGA. After it is processed by the host computer, it is directly sequenced to the ROLLS processor. Since the processing performed by the host computer could eventually be realized by the FPGAs, the hybrid system is considered equivalent to the emulation scenario from a strictly behavioral point of view, although the network response has a slightly longer delay. However, it is significantly simpler to implement and debug. Based on these considerations, the hybrid setup is deemed to be equivalent and thus, it is the preferred method for emulating the stereo network.

6.2 A re-configurable on-line learning spiking neuromorphic processor

The ROLLS neuromorphic processor is a full-custom, mixed-signal VLSI device with neuromorphic learning circuits that emulate the biophysics of real spiking neurons and dynamic synapses. This can be used to explore the properties of computational neuroscience models and to build event-based computing systems. The architecture is explained in Section 6.2.1 and the building blocks that are relevant in order to implement the stereo network are discussed in Section 6.2.2. Both sections were originally written by Qiao et al. (2015). In order to preface the section which follows, they are presented here for reference as originally published, with only minor changes.

6.2.1 The neuromorphic processor architecture

The block-diagram of the neuromorphic processor (NP) architecture is shown in Fig. 6.3. The device comprises a configurable array of synapse circuits that produce biologically realistic response properties and spiking neurons that can exhibit a wide range of realistic behaviors. Specifically, this device comprises a row of 256×1 silicon neuron circuits, an array of 256×256 learning synapse circuits for modeling long-term plasticity mechanisms, an array of 256×256 programmable synapses with short-term plasticity circuits, a 256×2 row of linear integrator filters denoted as *virtual synapses* for modeling excitatory and inhibitory synapses that have shared synaptic weights and time constants, and additional peripheral digital input and output (I/O) circuits for both receiving and transmitting spikes in real-time off-chip.

The NP was fabricated using a standard 180 nm CMOS 1P6M process. It occupies an area of 51.4 mm^2 and has approximately 12.2 million transistors. The die photo of the chip is also shown in Fig. 6.3. The silicon neurons contain circuits that implement a model of the adaptive, exponential integrate-and-fire (IF) neuron (Brette and Gerstner, 2005), post-synaptic learning circuits used to implement the spike-based weight-update/plasticity mechanism in the array of long-term plasticity synapses, and analog circuits that model homeostatic synaptic scaling mechanisms operating on very long time scales (Rovere et al., 2014). The array

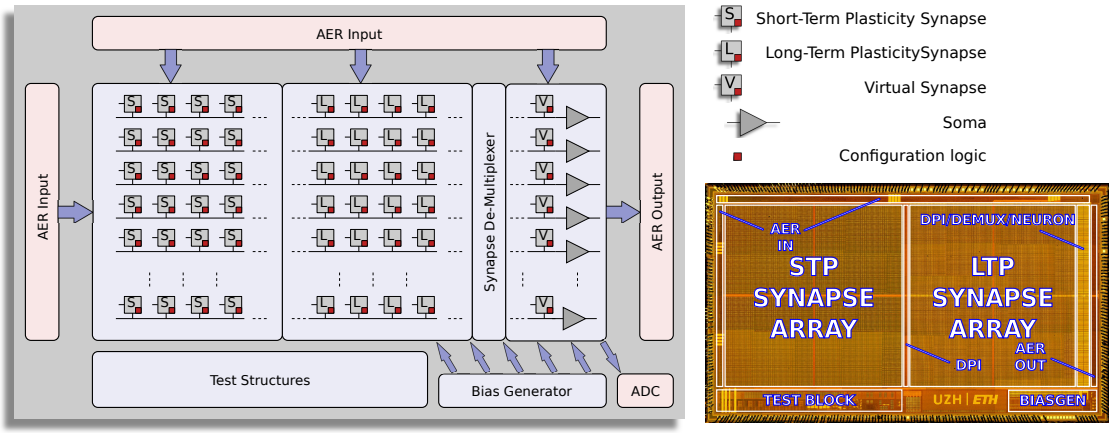


Figure 6.3: Architecture of the ROLLS neuromorphic processor. The block diagram of the architecture shows two distinct synapse arrays (short-term plasticity and long-term plasticity synapses), an additional row of synapses (virtual synapses) and a row of neurons (somas). A synapse de-multiplexer block is used to connect the rows from the synapse arrays to the neurons. Peripheral circuits include asynchronous digital AER logic blocks, an analog-to-digital converter, and a programmable on-chip bias-generator. A micro-photograph of the neuromorphic processor is shown on the bottom right. The chip was fabricated using a 180 nm CMOS process and occupies an area 51.4 mm^2, comprising 12.2 million transistors.

of long-term plasticity synapses comprises pre-synaptic spike-based learning circuits with bi-stable synaptic weights, that can undergo either long-term potentiation (LTP) or long-term depression (LTD). The array of short-term plasticity (STP) synapses comprises synapses with programmable weights and STP circuits that reproduce short-term adaptation dynamics. Both arrays contain analog integrator circuits that implement faithful models of synaptic temporal dynamics. Digital configuration logic in each of the synapse and neuron circuits allows the user to program the properties of the synapses, the topology of the network, and the properties of the neurons. The architecture comprises also a *synapse de-multiplexer* static logic circuit, which allows the user to choose how many rows of synapses should be connected to the neurons. It is a programmable switch-matrix that configures the connectivity between the synapse rows and the neuron columns. By default, each of the 256 rows of 1×512 synapses is connected to its corresponding neuron. By changing the circuit control bits, it is possible to allocate multiple synapse rows to the neurons, thereby disconnecting and sacrificing the unused neurons. In the extreme case all 256×512 synapses are assigned to a single neuron, and the remaining 255 neurons remain unused. An on-chip programmable bias generator, optimized for subthreshold circuits (Delbruck et al., 2010a) is used to set all of the bias currents that control the parameters of the synapses and neurons (such as time constants, leak currents, etc.). An analog-to-digital converter (ADC) circuit converts the subthreshold currents produced by selected synapse and neuron circuits into a stream of voltage pulses, using a linear pulse-frequency-modulation scheme, and transmits them off-chip as digital signals. Finally, peripheral asynchronous I/O logic circuits are used for receiving input spikes and transmitting output ones, using the AER communication protocol (Deiss et al., 1999; Boahen, 2000).

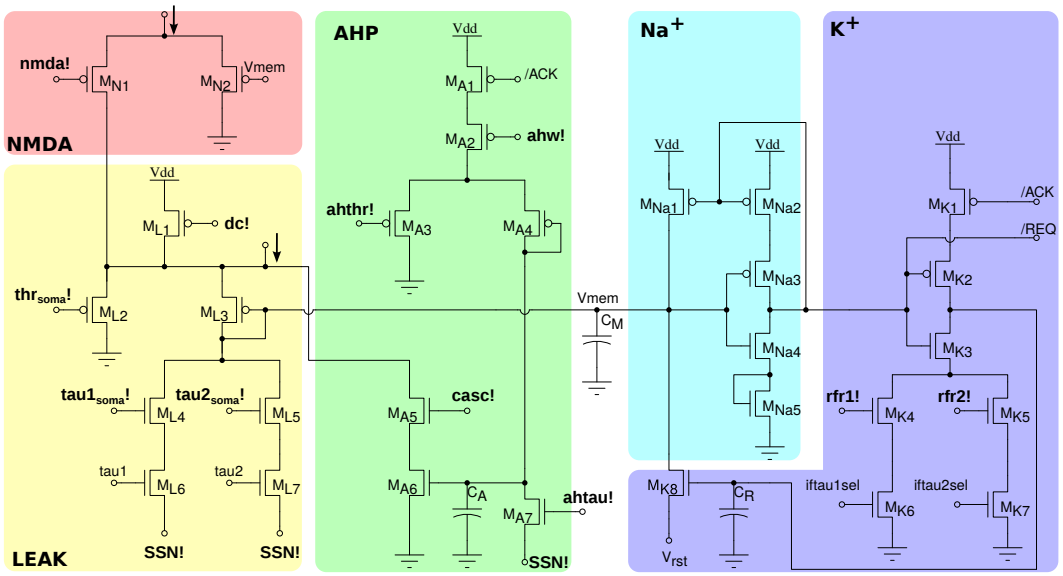


Figure 6.4: Silicon neuron schematics. The NMDA block implements a voltage gating mechanism; the LEAK block models the neuron's leak conductance; the spike-frequency adaptation block AHP models the after-hyper-polarizing current effect; the positive-feedback block N^+ models the effect of the Sodium activation and inactivation channels; reset block K^+ models the Potassium conductance functionality.

6.2.2 The neuromorphic processor building blocks

The main building blocks of the ROLLS chip that are used for the stereo network implementation are presented here. These blocks involve the neuron, the STP synapse array and the peripheral circuits.

The silicon neuron block

The neuron circuit integrated in this chip is derived from the adaptive, exponential IF circuit proposed in the work of Indiveri et al. (2011), which can exhibit a wide range of neural behaviors, such as *spike-frequency adaptation* properties, *refractory period* mechanism and *adjustable spiking threshold* mechanism. The circuit schematic is shown in Fig. 6.4. It comprises an NMDA block ($M_{N1,N2}$), which implements the NMDA voltage gating function, a LEAK circuit (M_{L1-L7}) which models the neuron's leak conductance, an AHP circuit (M_{A1-A7}) in negative feedback mode, which implements a spike-frequency adaptation behavior, an Na^+ positive feedback block ($M_{Na1-Na5}$) which models the effect of Sodium activation and inactivation channels for producing the spike, and a K^+ block (M_{K1-K7}) which models the effect of the Potassium conductance, resetting the neuron and implementing a refractory period mechanism. The negative feedback mechanism of the AHP block, and the tunable reset potential of the K^+ block introduce two extra variables in the dynamic equation of the neuron that can endow it with a wide variety of dynamical behaviors (Izhikevich and others, 2003). As the neuron circuit equations are essentially the same of the adaptive IF neuron model, we

refer to the work of Brette and Gerstner (2005) for an extensive analysis of the repertoire of behaviors that this neuron model can reproduce, in comparison to, e.g., the Izhikevich neuron model.

All voltage bias variables in Fig. 6.4 ending with an exclamation mark represent global tunable parameters which can be precisely set by the on-chip bias generator. There are a total of 13 tunable parameters, which provide the user with high flexibility for configuring all neurons to produce different sets of behaviors. In addition, by setting the appropriate bits of the relative latches in each neuron, it is possible to configure two different leak time constants ($\text{tau1}_{\text{soma}}$!, $\text{tau2}_{\text{soma}}$!) and refractory period settings (rfr1!, rfr2!). This gives the user the opportunity to model up to four different types/populations of neurons within the same chip, that have different leak conductances and/or refractory periods.

The short-term plasticity synaptic array

The array of short-term plasticity synapses contains circuits that allow users to program the synaptic weights, rather than changing them with a fixed on-chip learning algorithm. Specifically, each synapse has a two-bit programmable latch that can be used to set one of four possible weight values. In addition, it has an extra latch that can set the type of synapse (excitatory or inhibitory). In the excitatory mode, the synapse has additional circuits for modeling STD dynamics (Rasche and Hahnloser, 2001; Boegerhausen et al., 2003) whereby the magnitude of the EPSC decreases with every input spike, and recovers slowly in absence of inputs. Fig. 6.5a shows a block diagram of all synapse components while the schematic diagram of the synapse analog circuits is shown in Fig. 6.5b. In addition to the latches for setting the weight, there are two extra latches for configuring the synapse activation mode. There are three possible activation modes: direct, broadcast, and recurrent. In the direct activation mode the synapse is stimulated by an AER event that has the matching row and column address. In the broadcast activation mode the synapse is stimulated by an AER broadcast event (that has a dedicated address word) which targets the matching column address. All synapses belonging to the same column that have the BC_EN bit set high get stimulated in parallel, when the matching broadcast event is received. In the recurrent activation mode the synapse of column j is stimulated when the on-chip post-synaptic neuron of row j spikes. Therefore it is possible to connect, internally, neuron i to neuron j by setting the REC_EN bit high of the synapse in row i and column j. In addition to these circuits, there is a pulse extender circuit which can increase the duration of the input pulse from nanoseconds to hundreds of microseconds.

The left panel of Fig. 6.5b shows the excitatory current converter and the STD circuit. The current converter at the top generates a current that is proportional to the 2-bit weight. The proportionality constant is controlled through analog biases. This current charges up the C_{STD} capacitor through the diode connected p-FET M_{S3} so that at steady state, the gate voltages of M_{S1} and M_{W2} are equal. A pre-synaptic pulse on the PW port activates the I_{exc} current branch, and produces a current that initially is proportional to the 2-bit weight original current.

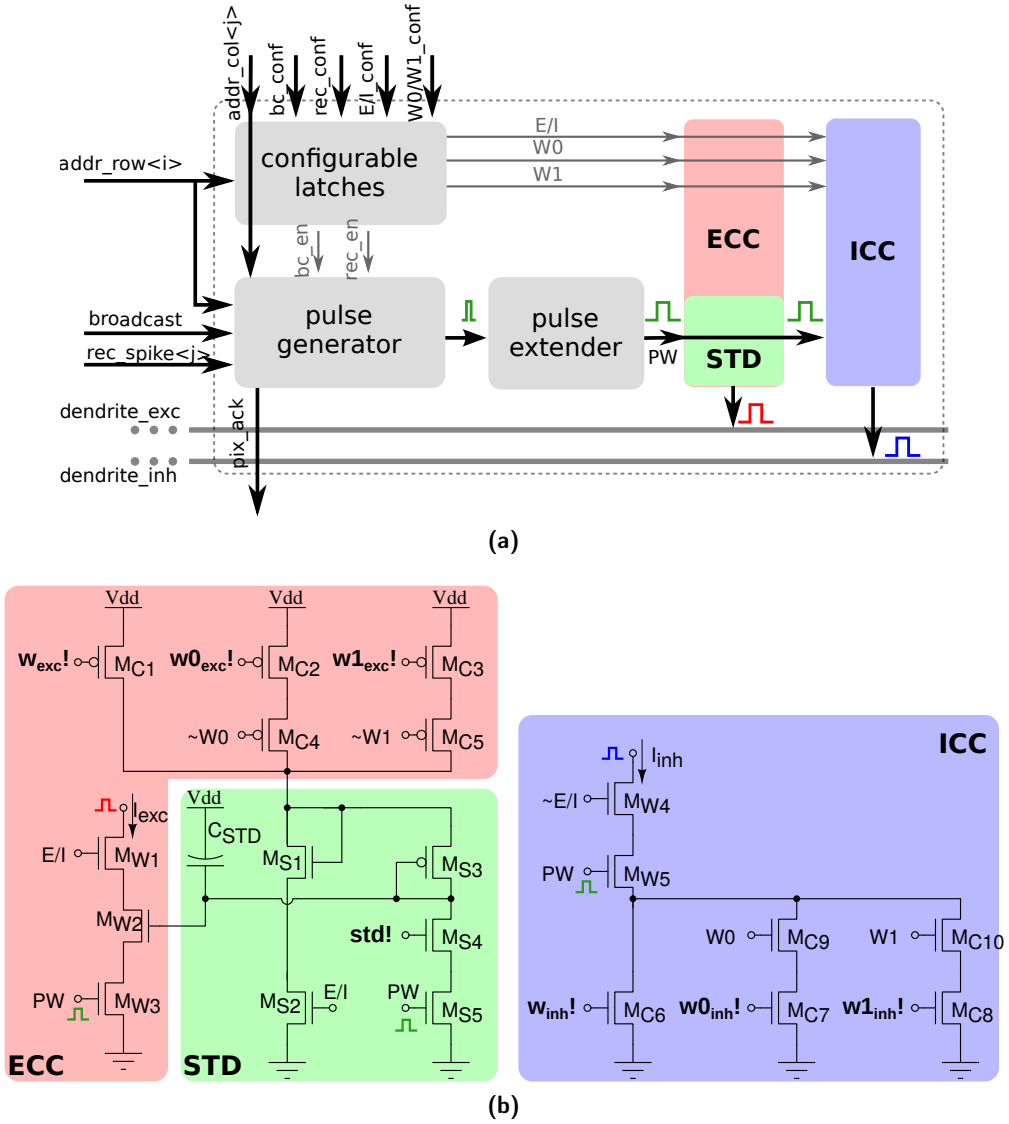


Figure 6.5: Short-term plasticity synapse array element. **(a)** Block diagram of the synapse element. **(b)** Transistor level schematic diagram of the excitatory and inhibitory pulse-to-current converters.

At the same time, the PW pulse activates also the STD branch through transistor M_{S5} and an amount of positive charge that is controlled by a bias (std!) is removed from the capacitor C_{STD}. The gate voltage of M_{W2} is now momentarily lower than that of M_{S1}, and recovers slowly through the diode connected p-FET M_{S3}. Pulses that arrive before the capacitor voltage has recovered completely will generate a current that is smaller than the original one, and will further depress the effective synaptic weight through the STD branch. The excitatory block is only active if the E/I voltage is high. If E/I is low, the inhibitory current circuit in the right panel of Fig. 6.5b is active and generates a weight-proportional inhibitory current on PW pulses.

The peripheral input and output blocks

The peripheral digital circuits are used to transmit signals into and out of the chip. Given the real-time nature of our system, we use *asynchronous* digital circuits and *quasi-delay-insensitive* circuit design techniques (Manohar, 2006) to avoid discretization or virtualization of time. The AER communication protocol used encodes signals as the address of the destination synapse or as a control word for the input side, and as the address of the sender neuron in the output circuits.

Input spike events as well as chip configuration events are sent through a common input interface that uses a 21-bit address space. Input addresses are decoded into a total of 1,249,553 possible patterns subdivided into three categories: *addressing, local configuration* and *global configuration. Addressing* inputs are decoded into a row and column address and are interpreted as a spike address-event, which are sent to the desired target synapse of a target neuron. *Local configuration* address-events contain the row and column address of the target element as well as extra configuration bits that are written to the local latches of the addressed element. *Local configuration* patterns include commands for setting the type of synapse, programming its weight, or enabling broadcast or recurrent connections. Finally, the *global configuration* inputs are decoded into configuration signals that represent global variables, stored onto registers in the periphery (rather than within the synapse or neuron elements). For example, the signals used to set the state of the synapse de-multiplexer are *global configuration* signals.

Each of the 256 neurons is assigned an 8-bit address for the output bus. When a neuron spikes, its address is instantaneously sent to the output AER circuits using the common four-phase handshaking scheme. Although neurons operate in a fully parallel fashion, their address-events can only access the shared output bus in a serial fashion. To manage possible simultaneous spike collisions the output AER circuits include an arbiter circuit that only grants access to the external bus to one neuron at a time.

6.2.3 The spike-response model

The effect of an input event on the membrane potential of a neuron can be mathematically described and is here termed “the spike response”. The system illustrated in Fig. 6.6 shows a block diagram of the spike-response model. An incoming event $\delta(t)$ is passed through the synaptic circuit (see Fig.6.5a) which produces a rectangular current pulse of duration ω and amplitude $\bar{I}_{syn}$ which is directly set by the bias currents controlling the synaptic weight (w_{exc}!, $w0_{exc}$!, $w1_{exc}$!). The post-synaptic current $i_{syn}(t)$ is then produced after the extended pulse is passed through a current-mode, low-pass filter (not shown in Fig. 6.5a but equivalent to the LEAK block in Fig. 6.4) earlier introduced as the *differential-pair integrator* (DPI) circuit (Bartolozzi et al., 2006). It can be either excitatory or inhibitory based on the configuration of the synapse. Finally, the current is summed in the soma circuit, where the *spike response* $i_{mem}(t)$ is produced.

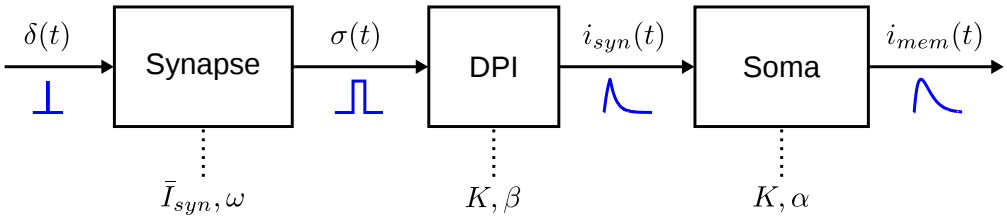


Figure 6.6: Block diagram of the spike-response model. The parameters $\bar{I}_{syn}$, α, β, ω and K determine the shape of the response. They are controlled by the bias settings of the associated circuits.

Under certain assumptions and if the adaption mechanisms of the synapse and neuron are disregarded, the entire transfer function can be approximated by a *second-order, low-pass filter* (see Appendix A.4 for more details). The spike response can then be derived as follows:

$$i_{mem}(t) = \begin{cases} K\left(1 - \dfrac{\beta e^{-\alpha t} - \alpha e^{-\beta t}}{\beta - \alpha}\right)\bar{I}_{syn} & t \leq \omega \\ \dfrac{K}{\beta - \alpha}\left[\beta e^{-\alpha t}\left(e^{\alpha\omega} - 1\right) - \alpha e^{-\beta t}\left(e^{\beta\omega} - 1\right)\right]\bar{I}_{syn} & t > \omega \end{cases} \tag{6.1}$$

where α, β, ω and K are control variables set by the bias parameters of the synapse (thr_{syn}!, tau_{syn}!, pw_{syn}!) and neuron (thr_{soma}!, tau_{soma}!). The spike response is maximal at the time t_{max}:

$$t_{max} = \frac{1}{\alpha - \beta}\ln\left(\frac{e^{\alpha\omega} - 1}{e^{\beta\omega} - 1}\right) \tag{6.2}$$

Evaluating the spike reponse at t_{max} yields the magnitude M:

$$M = i_{mem}(t_{max}) = K\left(e^{\beta\omega} - 1\right)\left(\frac{e^{\alpha\omega} - 1}{e^{\beta\omega} - 1}\right)^{\frac{\beta}{\alpha - \beta}}\bar{I}_{syn} \tag{6.3}$$

The duration T of the spike response is calculated from the beginning up until the point where its tail becomes negligible. This point can be set by a small threshold e, such that $i_{mem}(t \geq T) \leq e \approx 0$. Thus, the duration equals

$$T = \frac{e}{\min(\alpha, \beta)} + w \tag{6.4}$$

Although the magnitude of the spike response depends on the parameters α, β, ω and K, it is also proportional to $\bar{I}_{syn}$. This means that the bias settings for the synaptic weight are ideally suited to manipulate the magnitude, while the other parameters can be used to control the

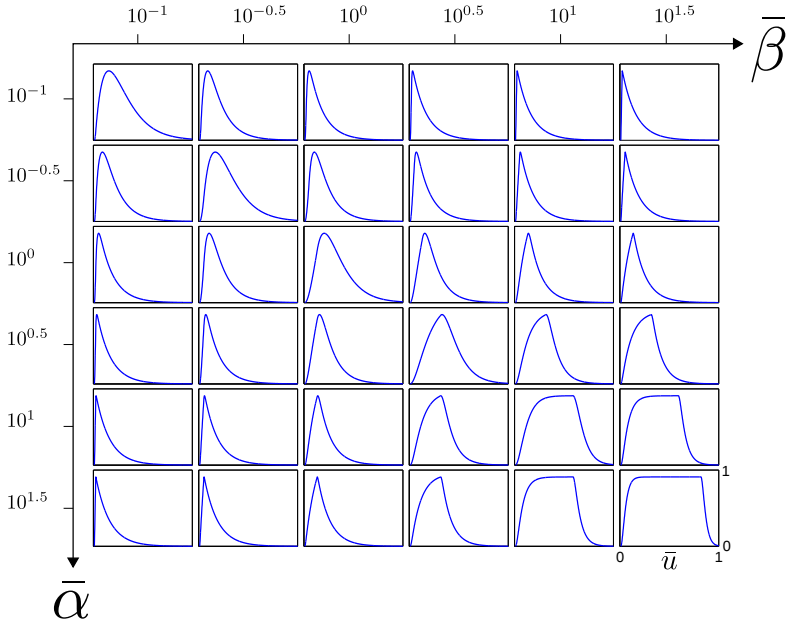


Figure 6.7: Different types of unit spike response kernels for varying $\bar{\alpha}$ and $\bar{\beta}$.

shape of the response. To study the effect of these parameters on the shape, the spike response is normalized by making the substitutions $\bar{\alpha} = \alpha\omega$, $\bar{\beta} = \beta\omega$ and $u = \frac{t}{\omega}$, which yields the *spike response kernel*:

$$\eta(u, \bar{\alpha}, \bar{\beta}) = i_{mem}(\omega u)\Big|_{\alpha=\frac{\bar{\alpha}}{\omega}, \beta=\frac{\bar{\beta}}{\omega}} \tag{6.5}$$

The *unit spike response kernel* can then be obtained by normalizing with the magnitude $\bar{M}$ and duration $\bar{T}$ of the spike response kernel:

$$\bar{\eta}(\bar{u}, \bar{\alpha}, \bar{\beta}) = \frac{1}{\bar{M}}\eta(\bar{u}\bar{T}), \qquad 0 \leq \bar{u} \leq 1 \tag{6.6}$$

Fig. 6.7 shows possible types of unit spike response kernels for varying levels of $\bar{\alpha}$ and $\bar{\beta}$. As the spike-response model is a linear system, the membrane potential response $I_{mem}(t)$ of the neuron to an arbitrary sequence of events is easily determined by superposing the individual spike responses of each event

$$I_{mem}(t) = \sum_i \eta_i \left(\frac{t - t_i}{\omega_i}, \alpha_i\omega_i, \beta\omega_i \right) \tag{6.7}$$

where t_i denotes the spike timing of the events. Note that different spike response kernels can be used simultaneously if the events are sent to different synapses (e.g. inhibitory, excitatory).

This is indicated by the dependency of the kernel η_i on the current event.

Applicability to stereo network emulation

The previously proposed stereo network model employs simple LIF neuron dynamics. In fact, these dynamics are described by the same differential equation as a first-order, low-pass filter. It can be demonstrated that the spike-response model as derived corresponds to a second-order, low-pass filter (see Appendix A.4 for more details). If the bias settings are chosen such that the temporal dynamics of the synaptic circuits are much faster than the soma dynamics, then the behavior becomes analogous to that of a first-order, low pass filter. This is the case when it is assured that $\alpha \ll \beta$. In addition, the width ω of the extended pulse should be kept relatively short to comply with the stereo model, i.e. $\omega \ll \frac{1}{\alpha}$. This is because in the case of the stereo model, spikes are represented by delta impulses. The LIF neuron instantaneously reaches its maximum in response to such a spike and exponentially decays thereafter. Similar responses are shown in the upper right corner of Fig. 6.7. Here, it can be directly observed that the condition $\bar{\beta} \gg \bar{\alpha}$ holds, which satisfies the first requirement that $\alpha \ll \beta$. The second requirement is also satisfied, as can be easily observed: $\bar{\alpha} \ll 1 \implies \alpha\omega \ll 1 \implies \omega \ll \frac{1}{\alpha}$.

6.3 Experiments and Results

6.3.1 Real-time parameter tuning

The full-scale network simulations presented in the previous chapter are computationally expensive when implemented on traditional Von Neumann architecture. Thus, they are hard to run in real-time which makes it difficult to explore network parameters. In a preliminary experiment, a scaled-down version of the stereo network is investigated in order to study and tune the *intrinsic* network dynamics in real-time. From the previous simulations, it was observed that the spiking stereo network is fairly robust and easy to tune. The effect of varying the size of the receptive field was examined and it was observed that the parameters of the neural models are not very critical, despite their obvious dependency on the speed at which the stimuli move. Although most of the network parameters are particularly dependent on the scene context, the experiments have shown that even in the case of scenes comprising of differing stimuli speeds (e.g. two people walking at different depths) no significant difference in performance can be observed within the scene. Nevertheless, it was suggested that the *sensitivity* of the disparity detector neurons is a critical parameter. For any given synaptic inputs and fixed time constant, the sensitivity of the neurons is set according to the *firing threshold*. The sensitivity determines the responsiveness to potential targets, whereby the current response controls the inhibition of the other neurons, ideally in order to suppress false targets. The real-time setup presented here is optimal for studying this mechanism in more detail.

Experimental setup

The experimental setup was used as described in section 6.1.4, which details the configuration of the *simulation*. The sensors where calibrated with a subpixel reprojection error (e_{RMS} = 0.249) using the technique described in Appendix A.2. The output of each sensor undergoes a homographic transformation, which is known as rectification. This transformation ensures that any arbitrary point in the field of view is projected onto the same vertical coordinates in the image plane of each sensor. In other words, this means that it is sufficient to search for corresponding matches along horizontal lines in the image plane, known as epipolar lines. The inset at the top of Fig. 6.8 shows the *rectified views* of both sensors with the central epipolar line highlighted (in yellow). In this setup, rectification is carried out in real-time without a significant increase in the CPU load. The monitors produce stimuli consisting of vertical bars moving in opposite directions. They were placed such that the range of both stimuli projects to a segment of the central epipolar line spanning 50 pixels in width in both image planes. Only these one-dimensional segments were considered as inputs to the stereo network, which narrows the coincidence and disparity detection layer down to a two-dimensional population of 50×50 neurons each.

Analysis of stereo matching performance

To assess the performance of the stereo network, a ground-truth model is required. The setup was designed such that subsequent trials reproduce the exact same visual stimuli. Therefore, ground-truth data can be easily measured by presenting only one stimuli at a time so that there is no ambiguity. A ground-truth model was then stored as a binary map, which encoded all of the true targets in disparity space, as shown in Fig. 6.9a.

Figs. 6.9b and 6.9c show histograms detailing the spikes of coincidence and disparity neurons in the stereo network after 660 trials. The PCM metric is used to access the matching performance. The coincidence and disparity spikes were compared with the ground-truth data and accordingly labeled, whereby the set of *correct coincidences* is denote by C^+, the set of *correct disparities* by D^+, the set of *false coincidences* by C^- and the set of *false disparities* by D^-. In addition to the PCM, two further metrics are introduced. The *true target amplification* (TTA) is defined as:

$$\text{TTA} = \frac{D^+}{C^+} \tag{6.8}$$

Accordingly, the *false target amplification* (FTA) can be expressed as follows:

$$\text{FTA} = \frac{D^-}{C^-} \tag{6.9}$$

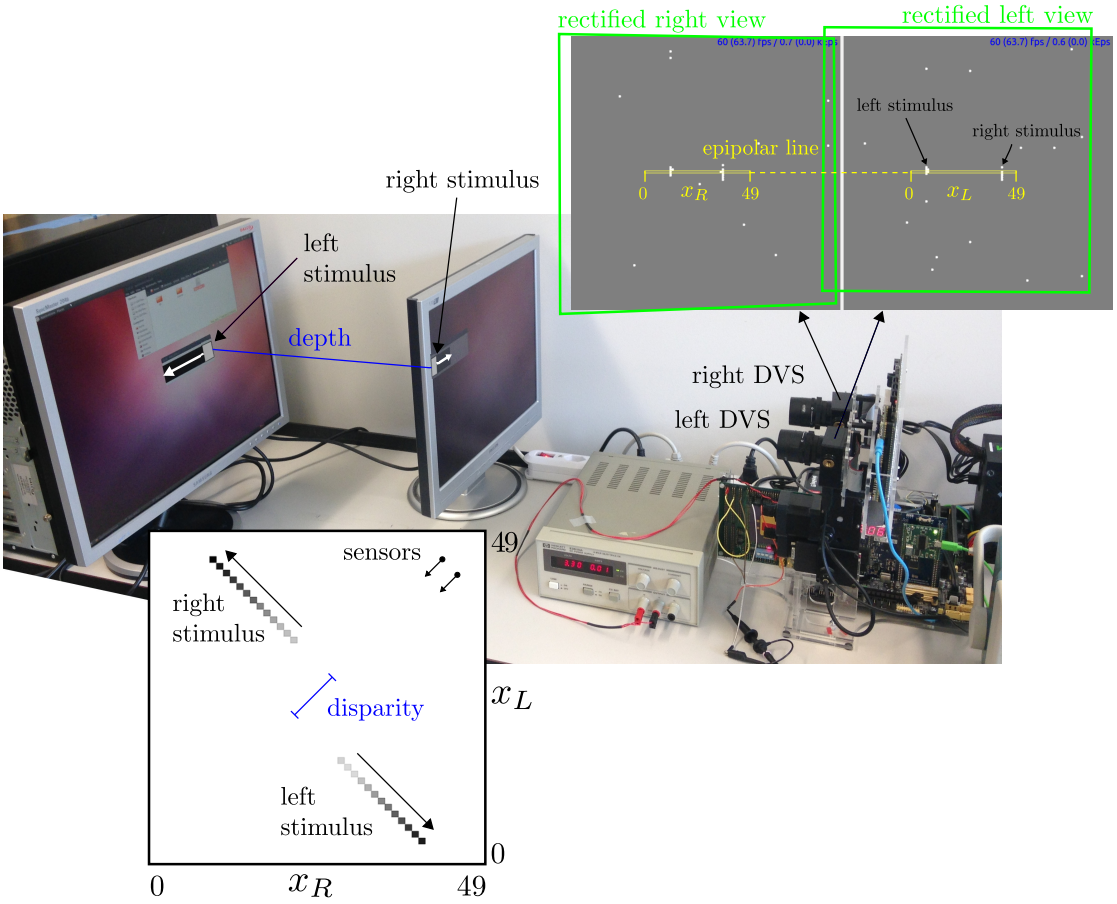


Figure 6.8: Experimental setup to tune the simulation of the stereo network in real-time. The real-time stereo setup is shown on the right while the two LCD monitors on the left are used for the generation of two moving stimuli that are separated in depth. The rectified output of the sensors is shown above, whereas the inset on the bottom shows the activity of disparity neurons.

Disparity neuron sensitivity and mutual inhibition

While the stimuli were continuously moving opposite to each other on their fixed trajectories, the firing threshold $\theta_{\mathbf{d}}$ of the disparity neurons was tuned in real-time. This enabled a better understanding of the critical relationship between the responsiveness of the disparity neurons, and the degree to which false targets are suppressed by mutual inhibition. When the sensitivity of the neurons is increased (by lowering $\theta_{\mathbf{d}}$), this effects not only their responsiveness to true targets (represented by the TTA) but also determines the degreee to which potentially false targets are inhibited, which in turn affects the PCM. To illustrate this causal chain, the PCM, TTA and FTA were recorded while sweeping $\theta_{\mathbf{d}}$. This was carried out once while mutual inhibition was disabled, and once in the normal case. A snapshot of the sub-threshold membrane potential $v_{\mathbf{d}}$ of disparity neurons is shown in Fig. 6.10a for both cases. Increased activity in regions where false targets are located can be clearly observed when inhibition is deactivated. The TTA and FTA are shown in Fig. 6.10b and the PCM is shown in Fig. 6.10c. In

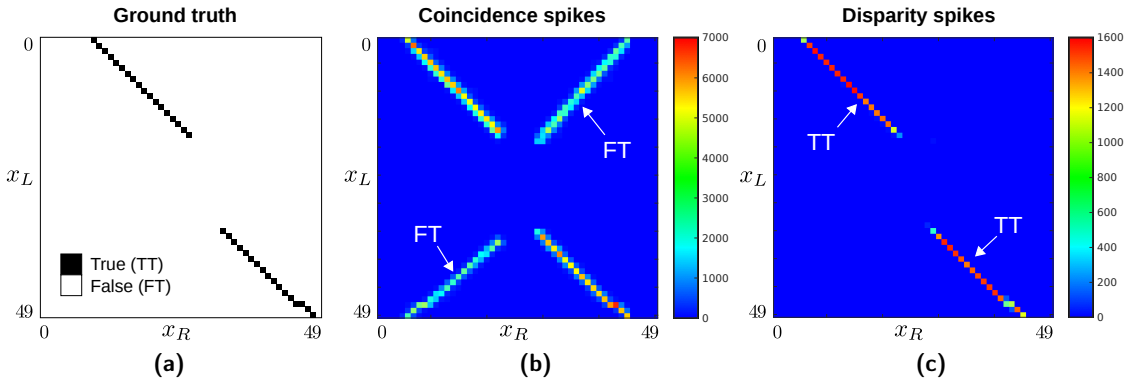


Figure 6.9: Results of the simulation of the scaled-down, real-time stereo network. **(a)** Disparity map of the ground-truth model of two moving stimuli (viewed from above). Both stimuli move at different but constant disparities. **(b)** Histogram of spikes from coincidence neurons. Due to ambiguity, two false targets (FT) can be observed which move perpendicularly to the real stimuli. **(c)** Histogram of spikes from disparity neurons. The stereo network successfully resolves the correspondence problem and encodes only true targets at the level of the disparity neurons.

the case where mutual inhibition is disabled, the PCM strongly depends on $\theta_{\mathbf{d}}$ as expected. Higher sensitivity (low $\theta_{\mathbf{d}}$) makes it more likely that false targets will go beyond the threshold and produce a spike. This is particularly apparent when inhibition is deactivated, as shown by the significant FTA in Fig. 6.10b. In contrast, when inhibition is activated, the FTA is almost perfectly suppressed over the entire range of $\theta_{\mathbf{d}}$. The TTA is nearly identical in both cases, meaning that the beneficial effect of mutual inhibition should be directly reflected in the PCM. As can be seen in Fig. 6.10b, this is exactly what happens. Only a slight decrease in the PCM is observed when sensitivity is increased and it remains close to 100% overall for a wide range of thresholds. This is very desirable as it suggests that disparity neurons can be very sensitive, without diminishing the matching performance of the network. This mechanism can be interpreted as a kind of *pseudo-normalization* which compensates for the fact that the neurons compute a covariance approximately, rather than computing a normalized correlation.

6.3.2 Simulating a real-time, spiking stereo network

For the sake of completeness, the real-time performance of a *simulation* of the full-scale stereo network is also assessed. The key variable under investigation is the maximum bandwidth (event rate) at which the simulation runs in real time. The simulation was implemented using C/C++ and embedded in the dedicated event-based stereo vision software framework. A distinct thread was assigned for executing the simulation code while the GUI, visualization and some preprocessing (e.g. stereo rectification) was handled by the main thread. It is worthwhile noting that no code optimization, other than on a behavioral level, was carried out. It is estimated that an optimized version of the code could yield about a five-fold improvement in bandwidth. Within the scope of this thesis, however, it is sufficient to restrict the investigation

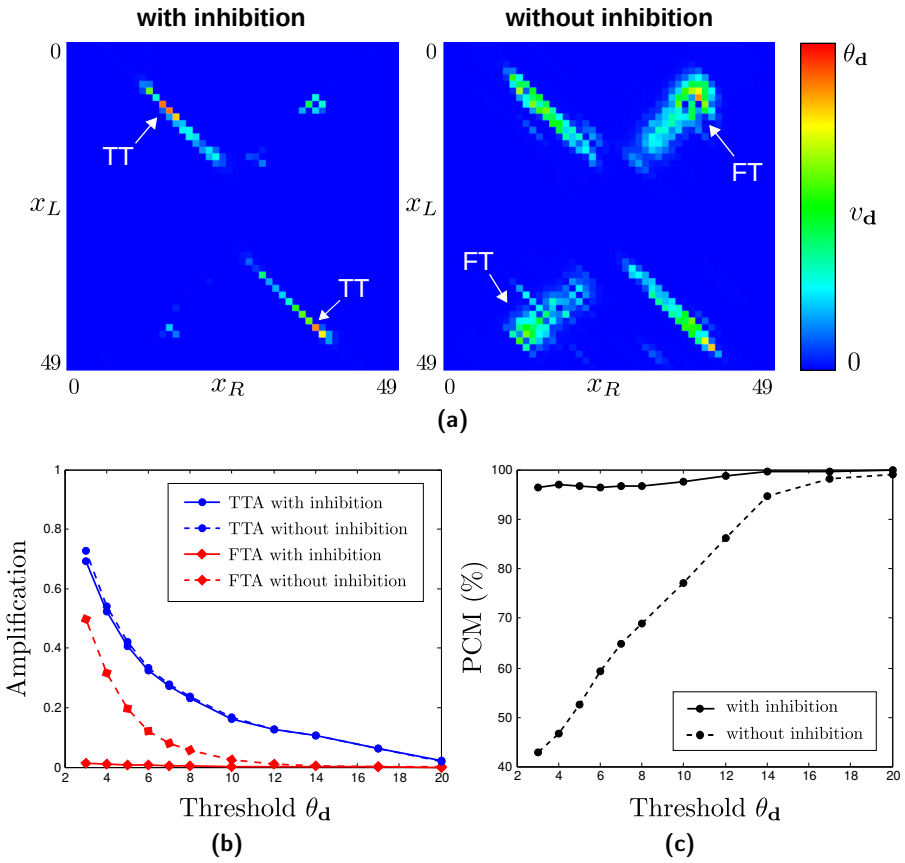


Figure 6.10: The effect of mutual inhibition in the down-scaled real-time stereo network simulation. **(a)** Subthreshold activity of disparity detectors for two scenarios, once with and once without mutual inhibition. Activity is represented by the membrane potential v_d. **(b)** TTA and FTA for the two scenarios. **(c)** PCM for both scenarios.

to the behavioral level. Indeed, even on this level alone, it quickly becomes evident that it would be currently impossible to implement such a massively parallel stereo network efficiently on a Von Neumann architecture.

Event-based lazy update approach

In a case where two event-based cameras have a spatial resolution of 128 × 128, the full-scale network incorporates a few million neurons. If this network were simulated in a classical manner, the physical entities would be synchronously updated at a high temporal resolution at discrete points in time, which would be computationally very expensive. As activity in the network is sparse, an obvious improvement would be to simulate only active units. However, it is even more beneficial to apply an approach referred to here as “event-based lazy update”, which can reduce the computational cost even further. The idea is based on two observations. Firstly, a LIF neuron can only spike at the time when it receives an input. Secondly, the state of the membrane potential can be easily determined at any time, if its state at an arbitrary

previous point in time is known, presuming that the neuron did not receive any input in the meantime. This means that all units can be simulated *asynchronously* and only have to be updated when they receive an input. Following on from the first observation, it should be noted that the firing events of coincidence neurons occur infrequently, depending on the timing of the input events. Thus, detecting firing events is relatively simple and computationally cheap. Similarly, coincidence events are handled in the same way as input events, meaning that disparity neurons are only updated when coincidences occur. This results in a simulation which can be said to be *truly* event-based (or data-driven) as polling data is not required throughout the entire simulation. The pseudo simulation code is detailed below.[1] It makes use of the *leaky-integrate-and-fire operation* (LIFOP), the pseudo code of which is also provided below. A LIFOP involves updating the neuron's membrane potential (given the last update time), adding an input, and checking the firing condition. The boolean value which is returned is either false, if the membrane potential remains below the firing threshold, or true, if it fires (whereupon the neuron is instantaneously reset to the resting potential).

Algorithm 2 Pseudo code for simulating the stereo network.

Require: Spikes e_i from event-based cameras $\mathcal{C}^L$ and $\mathcal{C}^R$

for all $e_i = (x_L, y, t)$ **do**
  for $d = d_{min} : d_{max}$ **do** ▷ *Iterate along line of sight*
    $r \leftarrow \text{LIFOP}(\mathbf{c}, +1)$ for $\mathbf{c} = (x_L, x_R + d, y)$ ▷ *Update* **c**
    if r = true **then**
      for $i = -\frac{\omega}{2} : \frac{\omega}{2}$ **do** ▷ *Iterate over receptive field*
        for $j = -\frac{\omega}{2} : \frac{\omega}{2}$ **do** ▷ *Excitatory receptive field*
          $r \leftarrow \text{LIFOP}(\mathbf{d}, +1)$ for $\mathbf{d} = (x_L + j, x_R + j, y + i)$ ▷ *Update* **d**
          if r = true **then**
            for $d = d_{min} : d_{max}$ **do** ▷ *Inhibit all neurons on both line of sights*
              $\text{LIFOP}(\mathbf{d}, -1)$ for $\mathbf{d} = (x_L + j + d, x_R + j, y + i)$ ▷ *Update* **d**
              $\text{LIFOP}(\mathbf{d}, -1)$ for $\mathbf{d} = (x_L + j, x_R + j + d, y + i)$ ▷ *Update* **d**
            end for
          end if
        end for
        for $j = -\frac{\omega}{2} : \frac{\omega}{2}$ **do** ▷ *Inhibitory receptive field*
          $\text{LIFOP}(\mathbf{d}, -1)$ for $\mathbf{d} = (x_L + j, x_R - j, y + i)$ ▷ *Update* **d**
        end for
      end for
    end if
  end for
end for

[1] For the sake of simplicity, the pseudo code only considers spikes from the left source. In the actual implementation, spikes from both sources are processed similarly.

Algorithm 3 Pseudo code for the leaky-integrate-and-fire operation (LIPOF).

```
1: function LIPOF(x, s)                          ▷ Input arguments x = (x, y, d) and s ∈ {−1; +1}
2:     δt ← t_now − t_last                       ▷ Time since last update
3:     v_x ← v_x(t_last) · exp(−δt/τ_x) + s      ▷ Add input s and update
4:     r = v_x > θ_x ? true : false              ▷ Check firing condition
5:     if r = true then
6:         v_x ← 0                               ▷ Reset
7:     end if
8:     return r
9: end function
```

Performance

As stated above, it is beyond the scope of this thesis to achieve the best possible real-time performance. However, it is still interesting to obtain an estimate of the complexity of implementing the stereo network on a Von Neumann architecture. For this purpose, complexity is measured in units of LIFOPs rather than conventional floating point operations (FLOPs). The total number of LIFOPs n for each input spike can be calculated as:

$$n = D + 2\alpha\omega^2 + 2\beta D \tag{6.10}$$

where D is the disparity range, ω the receptive field size and α and β are parameters that depend on the scene context and network parameters. The first term involves the LIFOPs of the coincidence neurons (which send a retinal input spike along a line of sight within the disparity range). The second term represents the LIFOPs of disparity neurons after a spike from the coincidence neurons. Note that the number of coincidence spikes that are triggered by each spike in retinal input depends on the scene context, which is determined by the amount of ambiguity. This variable is captured in α, whereby a value of $\alpha \approx 1$ would signify a simple scene with no ambiguity, such as a moving vertical bar. Conversely, a value of $\alpha \approx 5$ would signify a more complex scene consisting of multiple ambiguous objects such as a moving hand with five fingers. The final variable captures the inhibitory LIFOPs responsible for suppressing false targets. The number of disparity spikes for each spike in input is reflected by β and depends on the choice of network parameters. Recall that the output of the disparity neurons has a coarse cyclopean position. For this reason, it is combined with the coincidence spikes to yield the final output. This implies that there are more disparity spikes than output spikes and thus, that the network parameters should be chosen such that $\beta > 1$ in order to retain a uniform gain from the input and output of the network. The maximum achievable *real-time event rate* (RTER) can then be calculated simply by dividing n by the execution time t_{LIFOP} of a LIFOP. On an Intel Core i7-2600 CPU running at 3.40 GHz, this was experimentally measured as follows: $t_{LIFOP} = 50 \pm 5$ ns.

The RTER is shown in Fig. 6.11 for different scenes and receptive fields of varying widths. From

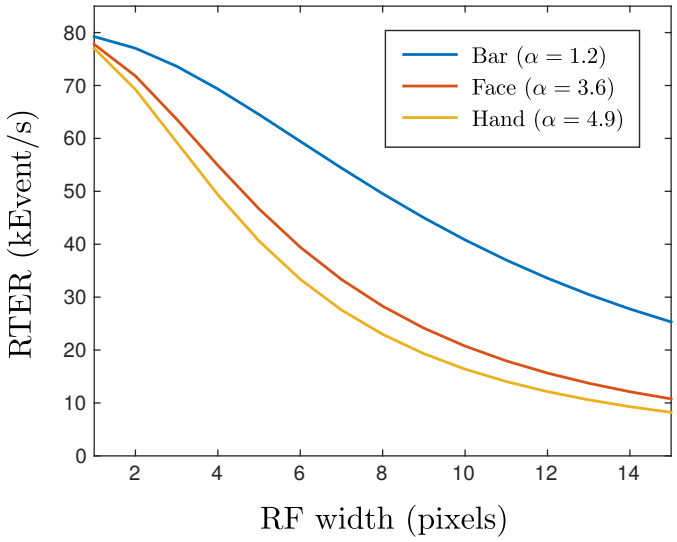


Figure 6.11: Real-time performance of the stereo network simulation for different scenes depending on receptive field (RF) width with $D = 50$ and $\beta = 2$.

previous experiments, it is generally known that if $\omega > 8$ it yields satisfactory results, which suggests an upper limit of about 50 kEvent/s for simple scenes and 25 kEvent/s for more complex scenes.[2] Interestingly, while it might be expected that the quadratic dependence on ω would have a dramatic effect, it does not dominate the other terms in the case of the most typical values. This holds true only if a wide disparity range is permitted, as is the case here ($D = 50$). If the disparity range would be further restricted, however, the cost of the quadratic term would become increasingly dominant.

6.3.3 Emulating a cortical disparity detector

The following experiment describes the complete process of emulating the single cortical disparity detectors in the proposed stereo network. In order to implement both the coincidence and disparity neurons, the analog neuromorphic circuits of the ROLLS processor are used.

Experimental setup

The experimental setup is schematically illustrated in Fig. 6.12. The stereo vision setup was assembled in the *hybrid* configuration. A single monitor was used to generate two vertically oriented bars moving in opposite directions in the same fronto-parallel plane of depth. In each retina, the events from two monocular receptive fields, each 9×9 pixels in size, were used as inputs. The receptive fields were centered at the retinal positions of true targets (T0 and T1). The retinal events were directly forwarded to the coincidence neurons on the ROLLS

[2] For simple scenes where there is little or no ambiguity, ω could be reduced which would result in a significant increase in bandwidth

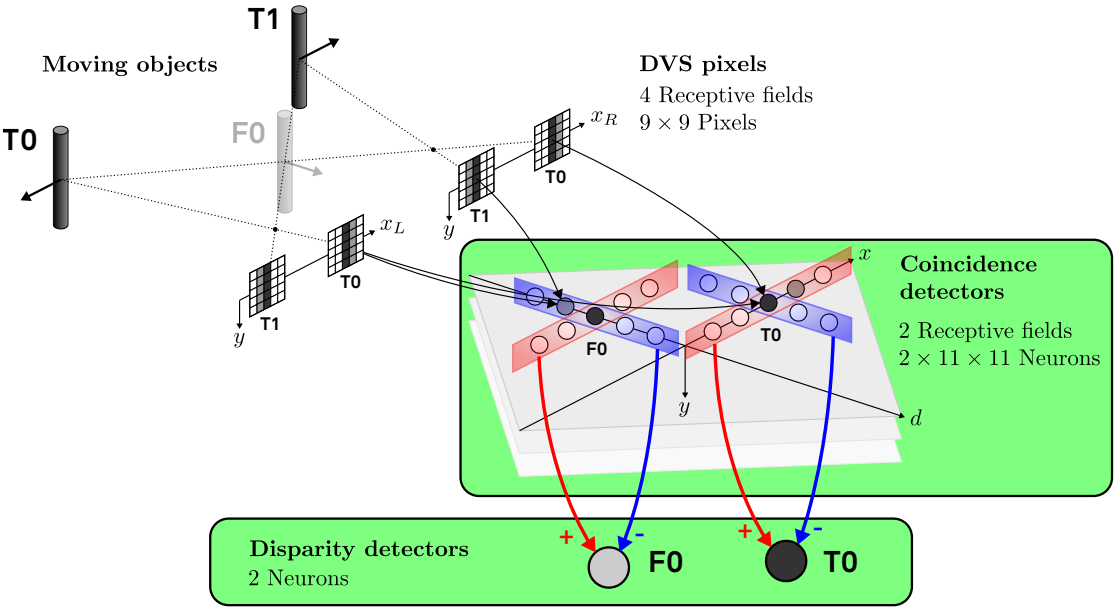


Figure 6.12: Experimental setup for emulating two disparity detectors. The coincidence and disparity neurons were emulated on the ROLLS neuromorphic processor as indicated by the green panels.

processor, whereby the digital routing and mapping were implemented on the host computer. As noted previously, this could have been completely implemented on the LGN FPGA without virtualization. In total, the 242 coincidence neurons (corresponding to a receptive field of $2 \times 11 \times 11$ in size) from a single type I disparity detector were implemented on the ROLLS processor, along with the disparity neuron itself. The disparity neuron was chosen such that it encoded the spatial location of one of the true targets (T0). In a second run, the mapping was reconfigured such that the disparity neuron encoded a false target (F0). The neuron response to the false and true targets were then recorded and compared.

Coincidence detection with mismatched neurons

The coincidence neurons were implemented as LIF models using the circuits provided by the ROLLS processor. It was observed that when adaption is disregarded, the behavior of the silicon neuron approximates a linear, second-order, low-pass filter. Furthermore, if the bias settings are chosen such that the time constant of the soma circuit is considerably longer than that of the synapse circuit and the extended pulse is short (see Section 6.2.3), the model reduces to a first-order, low-pass filter. As previously mentioned, this is equivalent to the sub-threshold model of the LIF neuron. The reduced circuit schematic, ranging from the synapse to the soma of a coincidence detector, is shown in Fig. 6.13a. The blue panel shows the pulse extender and excitatory synaptic weight circuit of the synapse, whereas the red panel shows the synapse's DPI circuit. Finally, the green panel contains the reduced soma circuit. An example of the measured membrane potential of a coincidence neuron is shown in Fig. 6.13b.

The red and orange bars indicate the spike times of an interocular pair of retinal cells. It can be observed that the coincidence neuron only spikes for a short *interocular temporal delay* (IOTD).

One of the major challenges associated with analog chips relates to the differing characteristics of schematically equivalent circuits, which is caused by *mismatched* transistor devices. This is due to the stochastic nature of the physical processes involved in fabricating such chips. Even a small mismatch in the effective size of a transistor can have a huge effect on the behavior of a complex circuit. Unfortunately, this is the case for the circuit described here, when it is used to emulate coincidence detection. A coincidence neuron involves only two of the 256 available synapses. Thus, the behavior of the circuit can be tuned by specifically selecting synapses. Fig. 6.14 depicts the variety of functional mismatches of coincidence neurons for all possible combinations of pairs of synapses. In other words, for each of the 256 neurons, all 32,786 combinations of synapses were tested and their sensitivity $S_{\Delta T}$ was recorded (recall that $S_{\Delta T}$ corresponds to the maximal IOTD at which the neuron spikes). In the scatter plot, each coincidence neuron is represented by a point indicating the mean $\mu(S_{\Delta T})$ and the standard deviation $\sigma(S_{\Delta T})$ of the sensitivity, both of which are computed across all combinatorial pairs of synapses for this particular neuron. The color indicates the percentage of feasible synaptic pairs, i.e. a pair of synapses that yielded a $S_{\Delta T}$ within the range of 0 to 20 ms. The crucial effect of mismatches on the presented coincidence detection circuit can be directly understood by examining the inset graph. The graph plots the quantity of neurons that have at least 10 feasible synaptic pairs for a specific $S_{\Delta T}$. In this experiment, the desired sensitivity of the coincidence neurons was 5 ms. The plot shows the result of the best tuning of manual bias after about 10 iterations. Only about 200 neurons are suitable coincidence detectors with $S_{\Delta T}$ =5 ms.

Disparity detector response to true and false targets

The circuit diagram of the disparity detector neuron is shown in Fig. 6.15. The soma integrates both excitatory and inhibitory currents from coincidence detectors in accordance with the model prescribed for the proposed stereo network. A different bias was used to differentiate between the excitatory weights of the synapses of coincidence ($w0_{exc}$!) and disparity ($w1_{exc}$!) neurons. For all the synapses (encompassing excitatory synapses from coincidence neurons, and excitatory and inhibitory synapses from disparity neurons) the biases for the DPI circuit have been set such that the dynamics are fast in comparison to the soma ($tau_{syn}! \gg tau_{soma}!$) in order to approximate the LIF neuron model. However, the dynamics of the soma regarding coincidence and disparity neurons must be different. The model in hand proposes that the integration time constants of the disparity neuron should be significantly longer. This can be implemented on the ROLLS processor by selecting different biases for the leak currents ($tau1_{soma}!$, $tau2_{soma}!$) of the soma circuit (see schematic in Fig. 6.4).

The spiking responses from the two disparity neurons were directly recorded from the AER output of the ROLLS processor. The tuning curves shown in Fig. 6.16 were obtained by repeat-

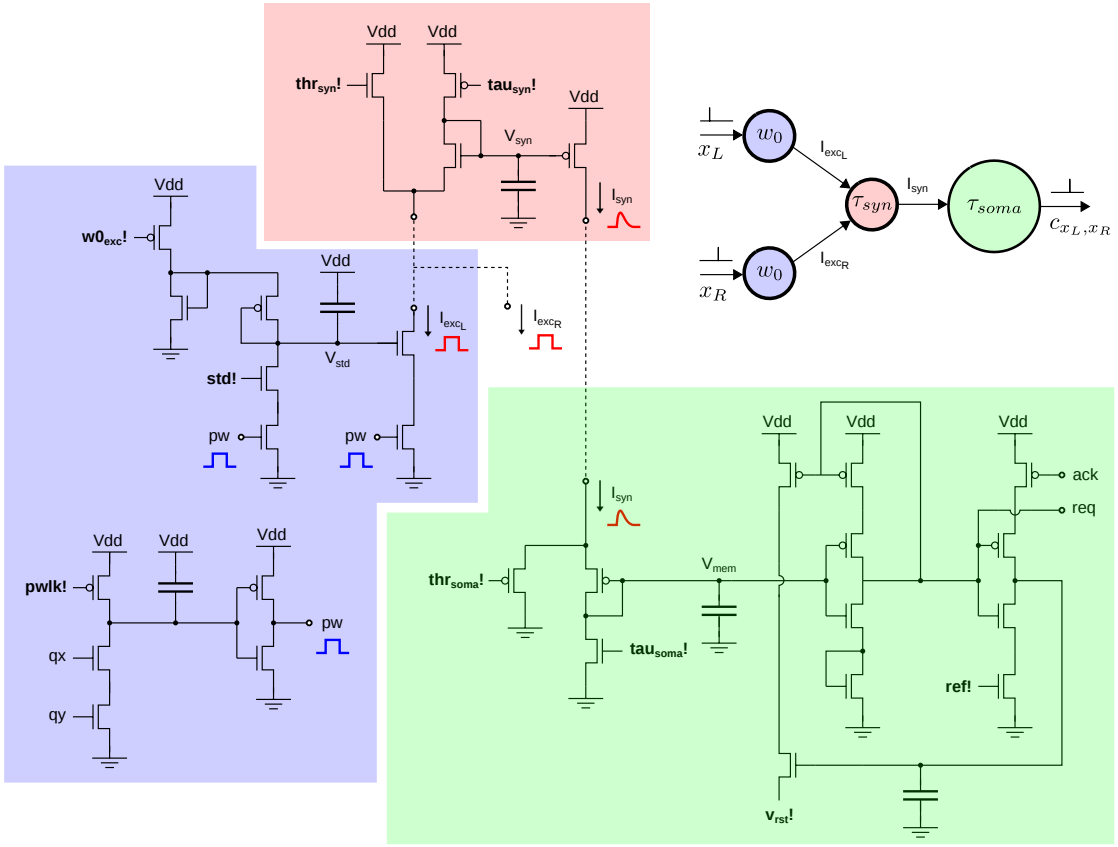


Figure 6.13: Emulating a coincidence detector. **(a)** Circuit schematic of the coincidence detector. The circuit contains two excitatory synaptic weight blocks (blue), a synaptic DPI block (red) and a soma block (green). The schematics here are reduced versions of the complete circuits shown earlier. An inset of the functional block diagram is shown on the top right. **(b)** Measured membrane potential (blue) of a coincidence detector neuron. Red and orange bars indicate spike times of input events. In this example the coincidence neuron was tuned to detect IODTs smaller than 10 ms.

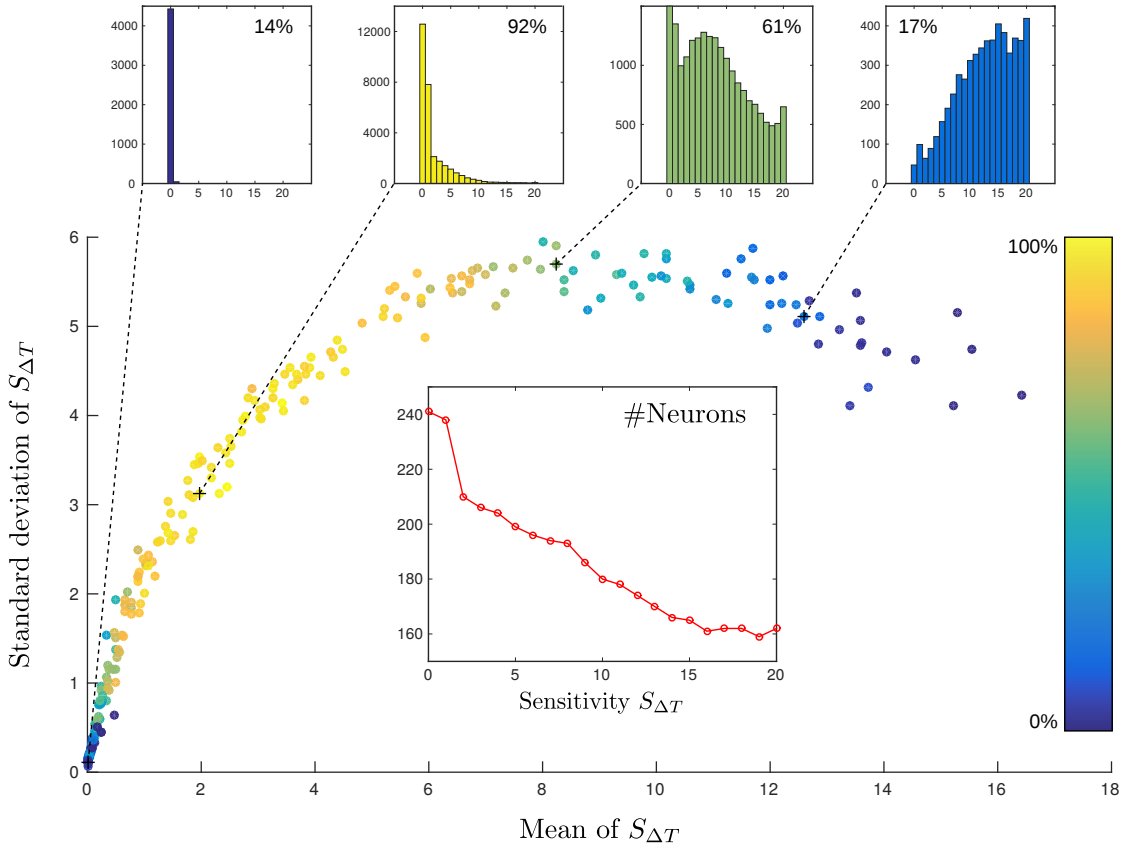


Figure 6.14: Behavioral variance of emulated coincidence detectors. Each point represents a coincidence neuron showing the mean and standard deviation of its sensitivity $S_{\Delta T}$ over all combinatorial synaptic pairs. The percentage of feasible synaptic pairs is color-coded. The histograms at the top show examples of distributions of $S_{\Delta T}$ for four selected neurons. The graph inset in the middle plots the quantity of neurons that have at least ten feasible pairs of synapses for a given $S_{\Delta T}$.

edly presenting stimuli, whereby the stimulus was slightly shifted in disparity and cyclopean position in each consecutive iteration. On the left side, the response of the disparity neuron at the position of the true target (T0) is shown, whereas the response of the neuron encoding the false target (F0) is shown on the right. The T0 response conforms with the archetypal characteristics of disparity-tuned excitatory cells; *narrow disparity selectivity* and *position invariance* (broad tuning in cyclopean position). Conversely, the F0 response resembles the behavior of tuned inhibitory cells. Interestingly, the selectivity to disparity and cyclopean position seems to be reversed here, as the response is suppressed for all disparities within the receptive field, but only at the preferred cyclopean position. This makes the neuron a very *robust* disparity detector.

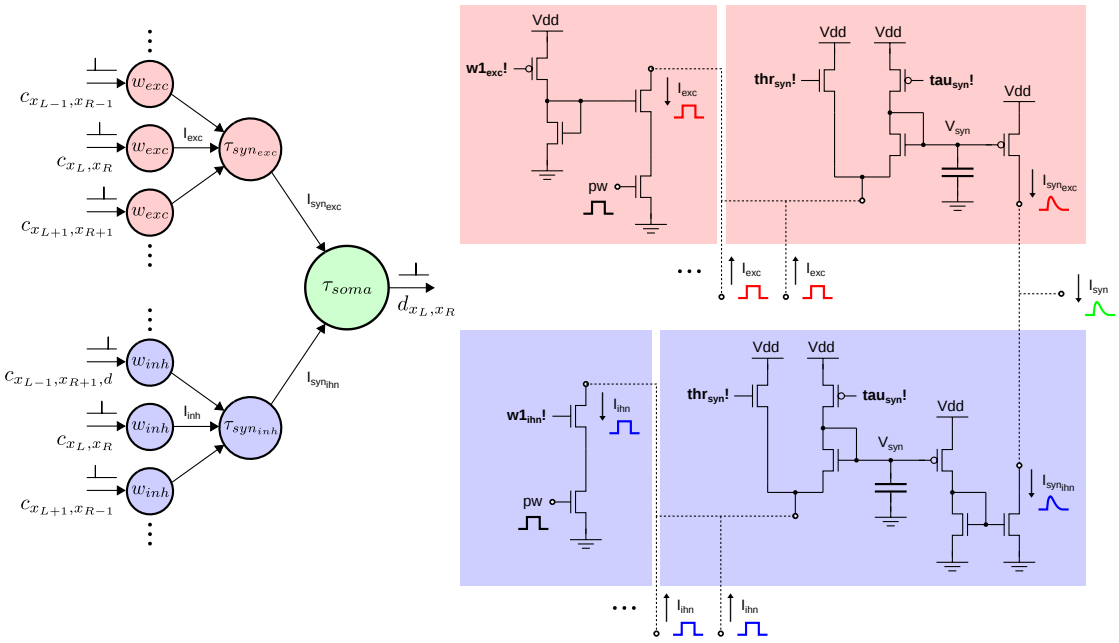


Figure 6.15: Circuit schematic of the disparity detector. The soma of the disparity neuron (circuit not shown) integrates excitatory (red) and inhibitory (blue) currents from coincidence neurons according to the receptive field map of the model of disparity neurons. The graph on the left shows that excitatory responses are obtained from coincidence neurons with constant disparity ($x_R - x_L = const.$), whereas inhibitory responses are obtained from neurons with constant cyclopean position ($x_R + x_L = const.$).

6.3.4 Emulating a spiking stereo network

The ROLLS processor was primarily designed for classification tasks, which typically require more synapses than neurons. The proposed stereo network, however, requires many neurons but only few synapses. For this reason, the ROLLS processor is not suitable for emulating a full-scale stereo network. Nevertheless, a proof of concept can be demonstrated by emulating an array of disparity neurons that represent a line of constant cyclopean position in disparity space. Such a network is able to detect the disparity of an arbitrary object moving through the fixation point of the silicon retinas.

Experimental setup

The experimental setup is illustrated in Fig. 6.17. An RDS stimulus was printed on a chart and moved at specified depths in a frontal plane parallel to the sensors. The retinal pixels, which were used as inputs, form 30 binocular receptive fields each of which is 21×21 pixels in size and are spread along the central epipolar lines of both sensors. These inputs project pairwise to 13,671 coincidence neurons, which stimulate the disparity neurons representing 30 disparities equally distributed along a line of constant cyclopean position. Once again, the stereo setup was configured in the *hybrid* mode. The disparity neurons were emulated

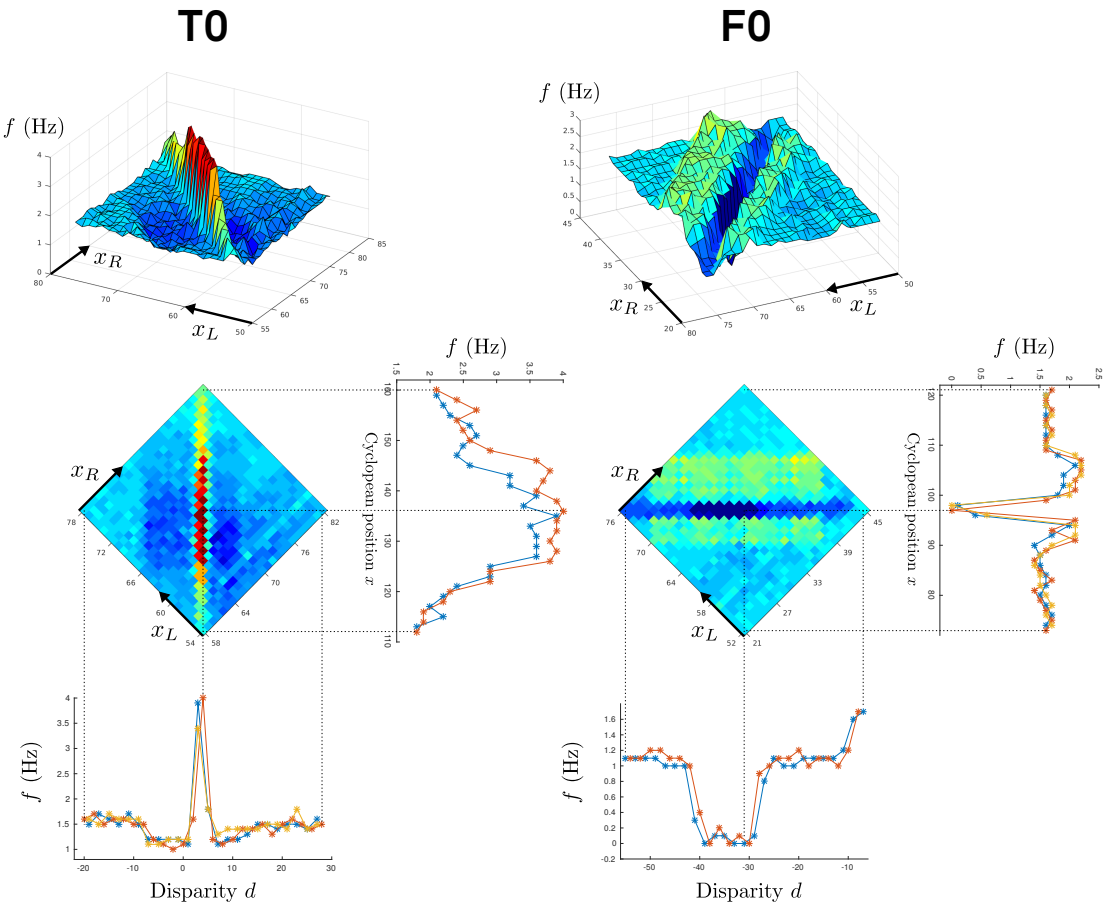


Figure 6.16: Responses from two emulated disparity detectors encoding a true (T0) and false (F0) target. The T0 response exhibits the typical behavior of similarity to tuned excitatory cells, whereas the F0 response seems to be reversed. At the top, the responses to various stimuli positions x_L and x_R in the left and right retina are shown as 3D surfaces. Below, the color-coded footprints are shown. Profiles of the responses along the axis of disparity ($d = x_R - x_L$) and cyclopean position ($x = x_R + x_L$) are shown on the sides of the footprints. T0 profiles are shown for two disparities ($d = 3, 4$) and three cyclopean positions ($x = 135, 136, 137$). F0 profiles are shown for three disparities ($d = -30, -31, -32$) and two cyclopean positions ($x = 97, 98$).

on the ROLLS processor while coincidence detection was *simulated* on the host computer. Mutual inhibition was implemented among the disparity neurons using on-chip recurrent connections. This model slightly deviates from the proposed stereo network, in which mutual inhibition occurs along lines of sight rather than cyclopean position. The resulting behavior can be interpreted as a winner-takes-all (WTA) mechanism, which is similar to the original model in that it also enforces the uniqueness constraint.

Disparity response of emulated stereo network

The response of the stereo network is shown in the form of a spike raster plot in Fig. 6.18b. The stimulus was presented at equally spaced disparities ranging from −10 to +10. The spikes

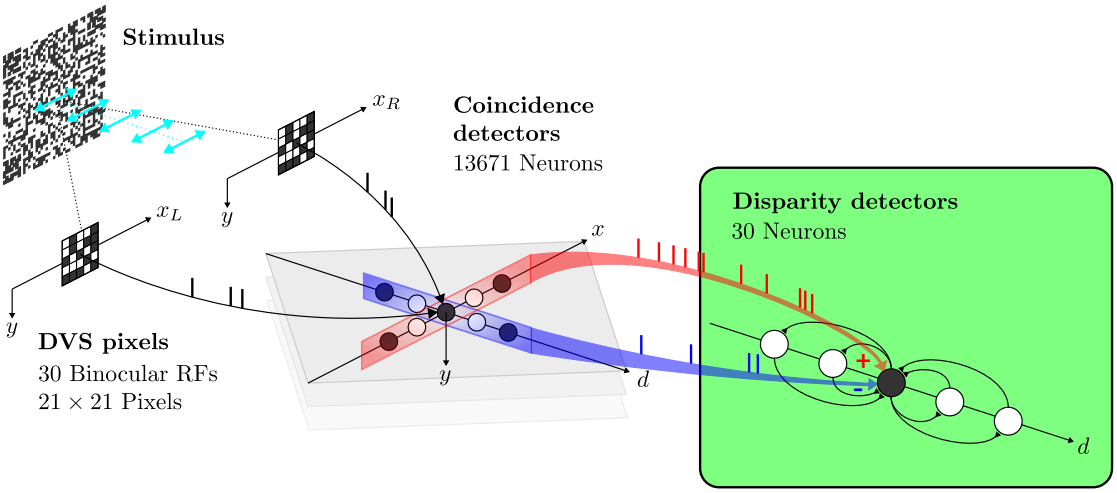


Figure 6.17: Experimental setup for emulating the stereo network. The coincidence neurons were *simulated* on the host computer, whereas the disparity neurons were *emulated* on the ROLLS processor (indicated by the green panel).

from the neuron encoding the true disparity are colored red, while all the others are colored blue. The network successfully detects the correct disparity of the stimulus in the entire range, as indicated by the red spike trains. While a certain disparity detector is active, the others are strongly suppressed by the WTA mechanism. In fact, there is quite some ambiguity present in the stimulus. This becomes apparent if the recurrent connections that are responsible for the WTA mechanism are deactivated. The network response in this scenario is shown in Fig. 6.18a for comparison. It can be observed that for all disparities, ambiguity induces significant activity.

6.4 Applications

The machine vision applications of today are entirely *frame-based*. However, the advantages of event-based cameras are clearly evident. In particular, event-based, neuromorphic stereo vision systems have huge potential to provide more efficient ways of implementing many applications. The following section summarizes areas of applications, discusses their requirements, and suggests new potential approaches.

6.4.1 Scope of event-based stereo vision

Event-based stereo vision is generally not suited to dense and high-resolution depth perception because of the nature of event-based cameras. Instead, a variety of frame-based approaches provide very satisfactory results (see Section 2.4). Many applications, however, do not require such detailed depth information. The advantages of neuromorphic stereo vision stem from the neuromorphic sensors. Therefore, a neuromorphic approach becomes inter-

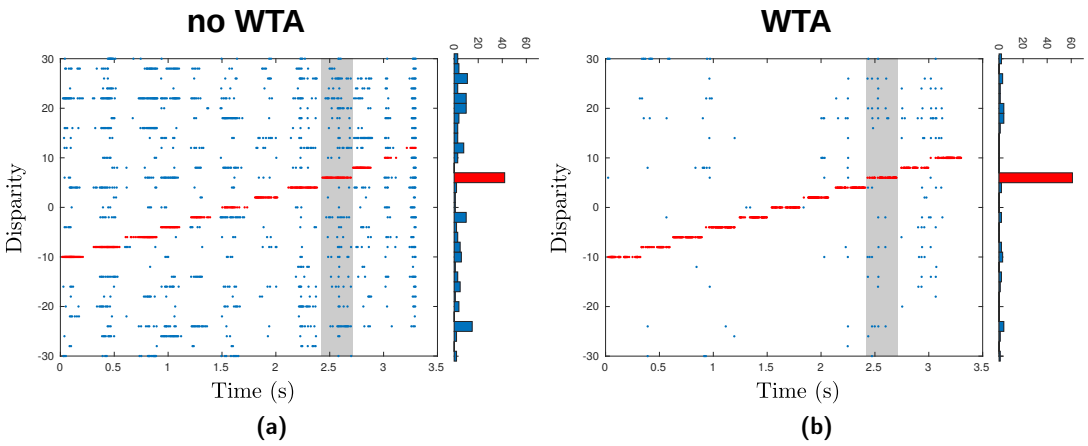


Figure 6.18: Spike raster plot of the response of the emulated stereo network. **(a)** Response of the stereo network without WTA. **(b)** Response of the stereo network with WTA. Examples of spike histograms for a given disparity are shown on the right of each plot. The histograms were obtained from the spikes within the gray region.

esting when *low latencies*, a *high dynamic range* or a *low power* budget is required. However, event-based stereo vision does not exhibit lower latency, less power consumption and higher dynamic range in all circumstances. A normal high-speed camera (>1000 Fps) could provide depth information with lower latency, a tiny ASIC for stereo vision could be more power efficient, and the use of high dynamic range (HDR) imaging techniques could provide higher dynamic ranges. In the case where a trade-off between these features is required, however, event-based stereo vision potentially offers a more efficient approach. The table below aims to list all potential applications based on their requirements and proposes a suitable event-based approach which would be expected to be more efficient than current approaches.

6.4.2 Feasibility study: An event-based stereo vision system for truly immersive virtual reality

The challenge of virtual reality (VR) is latency. Humans rapidly perceive a virtual scene to be unrealistic if the latency is not low enough. This is due to the discrepancy between the delayed visual information and inertial information from the inner ear. Often, this lag can even lead to motion sickness. A VR experience is deemed "truly immersive" when the human brain accepts that it is realistic. The magic number of the latency threshold for truly immersive VR is believed to be in the range of 5 to 20 ms (Regan et al., 1999; Adelstein et al., 2003; Mania et al., 2004). This imposes difficult constraints on the system that tracks the observer's head, which is required for any form of VR. To give an example, if a visual tracking system based on a normal 60 Fps camera is used, the lag introduced by the frame-rate is already 16.7 ms. To make things worse, a significant additional lag is caused by the process that estimates the pose from the captured images, and the process that renders the virtual

Table 6.1: List of applications for event-based stereo vision. The requirements involve latency (LAT), power consumption (PW), dynamic range (DR), resolution (RES) and hardware cost (COST). Requirements are rated very high (++), high (+), marginal (-) and low (–). The two main approaches are the spatiotemporal correlation (STC) stereo algorithm and the spiking stereo neural network (SSNN).

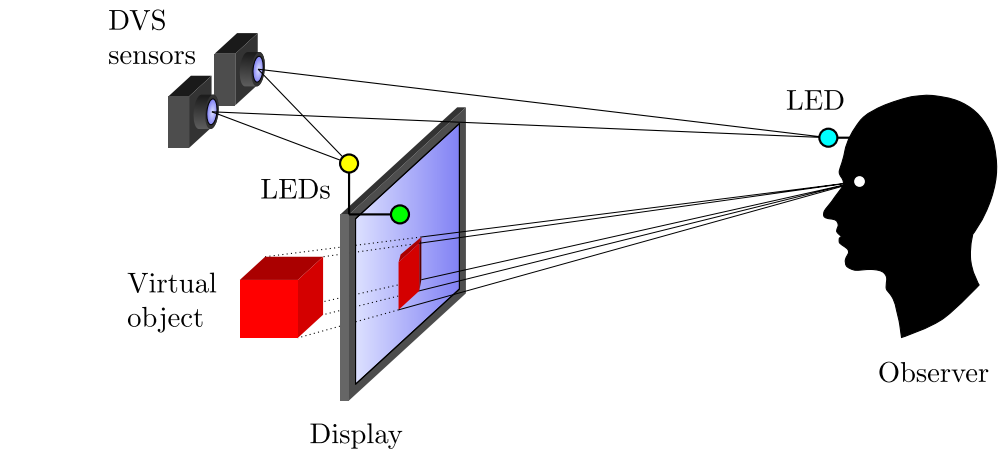
APPLICATION	DESCRIPTION	REQUIREMENTS					APPROACH
		LAT	PW	DR	RES	COST	
Robotics	Sense-and-Avoid and visual positioning for mobile robotic platforms (e.g. drones)	++	+	+	+	+	Large-scale SSNN implemented on custom mixed digital/analog ASIC, multi-chip setup (sensors and processor) Advantage: Latency and power consumption tradeoff
AR and VR	Augmented and Virtual Reality for smart glasses, mobile phones, tablets and headsets	++	++	+	+	++	Large-scale SSNN implemented on custom mixed digital/analog ASIC, preferably sensors and processor combined on SoC Advantage: Latency and power consumption tradeoff
Automotive	Advanced Driver Assistance Systems (ADAS)	++	–	++	+	-	STC algorithm implemented on powerful digital computing platform involving CPUs, FPGAs, GPUs and DSPs Advantage: Latency and dynamic range tradeoff
Range finder	Obstacle avoidance in robotics	+	+	-	–	+	Mini-scale SSNN implemented on a general neuromorphic processor Advantage: Passive and cheap
Entertainment	Motion and gesture controlled user input devices	+	–	-	+	++	STC algorithm implemented on powerful digital computing platform Advantage: Latency and cost tradeoff, fixated sensors reduce computational cost also for high sensor resolutions
Motion capture	Motion capture systems for sports and video and game production	-	–	-	++	++	Active, blinking LED markers, trivial matching problem Advantage: Very cheap, no correspondence problem, very low system requirements

scene on a display. Inertial measurement units (IMUs) have a very low latency (down to 1 ms) and they can yield an estimate of the pose with very little data processing. However, because IMUs measure accelerations and angular velocities, they are prone to significant drift errors compared to visual sensors that can measure position and orientation (meaning that they have zero drift). Given these circumstances, event-based cameras seem to be the ideal solution for VR applications. This feasibility study demonstrates the potential applicability of an event-based stereo-vision tracking system to VR.

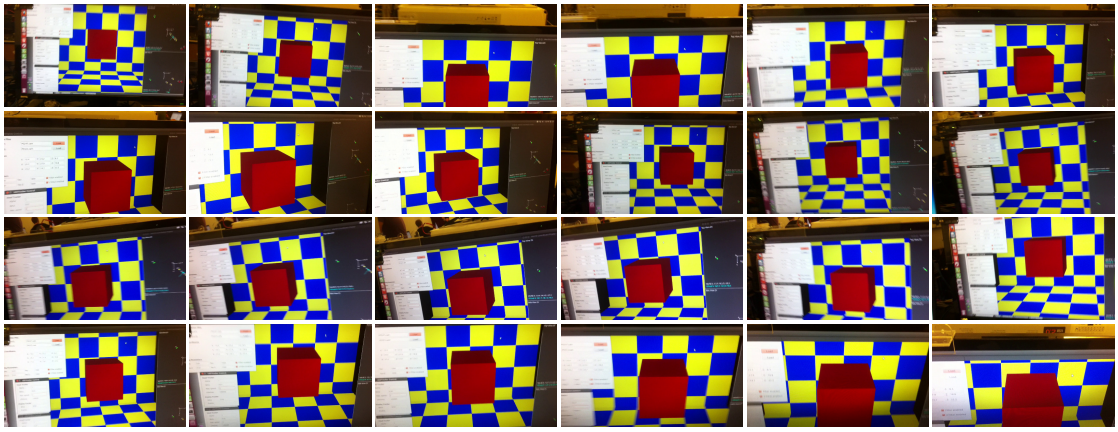
With the launch of Nintendo's Wii Remote in 2006, a new-generation virtual reality device became available. Lee (2007) used a head-mounted Wii Remote to accurately track an observer, rendering virtual scenes on a screen depending on the pose of his/her head. The virtual scene is then perceived as seen through a window, creating a realistic illusion of depth. The work of Lee (2007) served as inspiration for the idea presented here. A virtual environment is rendered on a monitor as seen through a window, based on the head position of the observer. The head position is tracked with an event-based stereo vision system placed slightly above and behind the monitor, such that the monitor's pose can also be tracked. To reduce the lag introduced by data processing, the correspondence problem is eased by using active LED markers. Each LED is programmed such that it blinks at a high but distinctive frequency (typically around 1 KHz). The emitted light pulses generate easily identifiable spike trains in both cameras, thus avoiding the need to deal with the correspondence problem. In total, three LEDs track the pose of the monitor and one head-mounted LED tracks the viewing position. An illustration of the entire setup is shown in Fig. 6.19a.

Although the lag of the entire system was not measured, the test subjects reported a realistic user experience. A demo video can be found online[3]. A series of snapshots from the video showing the rendered virtual scene displayed on the monitor at different viewing angles is given in Fig. 6.19b. Indeed, for this application, stereo vision is not strictly necessary. An alternative approach would be to mount the DVS sensor on the head of the observer (e.g. on a pair of 3D shutter glasses) and track the monitor. This approach would have a couple of advantages. One advantage is that only one sensor would be required because the relative pose of the head could be inferred from the projected view of the monitor, if its geometry is known. Another advantage is that the monitor would always be within the field of view because the camera could be aligned with the view of the observer. The tracking of the monitor could be achieved either with active LED markers or by passively tracking the frame of the monitor. Alternatively, the pixels themselves or the background flickering of the monitor could be used as a tracking source. Furthermore, a fast feedback loop could register the rendered scene of the projected view based on an estimate of the relative pose of the observer.

[3] https://www.youtube.com/watch?v=w7vfJ3VqBiQ



(a)



(b)

Figure 6.19: Immersive virtual reality with an event-based stereo vision system. **(a)** The virtual object is rendered on a display as if seen through a real window. The head and monitor position are continuously tracked with active LED markers and an event-based stereo vision system. **(b)** Series of images from the view of the observer as he moves his head.

6.5 Discussion

In this chapter, neuromorphic stereo vision systems on various physical substrates were investigated. In this regard, a major factor to consider is whether the substrate is a *simulating* or an *emulating* platform. The line between simulation and emulation is not clear-cut — indeed, it is quite a smooth transition. In this context, simulation is associated with the use of *general purpose* hardware, *serial* processes, *synchronous* communication, *virtualization* and *discretization*, whereas emulation is associated with *dedicated* hardware, *parallel* execution, *asynchronous* communication and real-world *physical* quantities that model themselves. The transition is said to be smooth because some of these characteristics are not clear-cut or can be mixed. The stereo network presented in this work employs many simple units that all work in parallel. It stands to reason, therefore, that the best results would be achieved

with a substrate that *emulates* all of these units. Indeed, an ideal emulation comprises a physical device for every unit of the network, yielding the lowest latency possible. In this scenario, the power consumption would be minimal since all signals propagate in real-time and do not have to be accelerated (as would be the case for a simulation). A proof of concept towards such an emulation was demonstrated in this chapter. However, the ROLLS processor which served as the physical emulation substrate is not nearly powerful enough to emulate the *full-scale* stereo network. Unfortunately, this is a general problem of neuromorphic hardware. The emulating units of neuromorphic devices are limited due to the fact that they are implemented on flat substrates, which can consequently only be scaled in two-dimensions. The existing large-scale neuromorphic platforms often apply principles of discretization and multiplexing (e.g. TrueNorth), or make even more steps in the direction of simulation (e.g. SpiNNaker). These devices are still more efficient than the extreme case of simulation by multiple orders of magnitude (e.g. single CPU) but clearly deviate from the extreme emulation scenario. The extent to which the power of the stereo network can be unleashed depends on the degree to which it can be emulated. In the extreme case of a pure simulation, it has been shown that it lags far behind existing frame-based approaches. This does not mean that an optimized and simplified GPU implementation might not yield comparable results, however. Nevertheless, GPUs are optimized for fast processing of synchronous frames. Thus, the stereo network would need to be modified and brought into a more synchronous form, possibly following a similar direction to the proposed efficient version of the STC algorithm discussed in Chapter 4. In contrast to large-scale networks, smaller stereo networks could be fully emulated on neuromorphic hardware. There are a few applications where such small-scale neuromorphic stereo vision systems could provide a very efficient and economically superior alternative to existing approaches.

6.5.1 Analog versus digital

A fundamental debate in neuromorphic engineering concerns the question of whether to use analog or digital circuits to emulate brain-inspired circuits. Again, this is closely related to the difference between simulation and emulation. True emulation is achieved using analog circuits that employ physical quantities that model themselves. For example, a synaptic current would be represented by a real current in the electrical circuit. Analog circuits are prone to the problem of device mismatch. Thus, they are often hard to tune and impractical. Conversely, digital circuits use the concept of discretization and thus, they can model the value of a synaptic current with multiple electrical charges (know as bits), for example. Although the performance of digital circuits is robust, they require more devices and faster signals which can result in a higher power consumption. The experiments have shown that the mismatch of the ROLLS processor has a dramatic effect on the sensitivity of coincidence neurons, while the behavior of the disparity neurons seems to be less variable. Of course, this is related to the fact that coincidence neurons integrate the input from only two units. Conversely, disparity neurons benefit from the averaging effect of the integration of many units. This suggests that a simple digital process to detect coincidences and an analog process to integrate

evidence (disparity detection) would be more suitable. More details on the prospect of a *mixed analog-digital* architecture are provided in the last chapter.

6.5.2 Unsupervised learning of epipolar constraints

One important requirement for the stereo network is that inputs are rectified. If the calibration of the sensors is known, the input can be easily rectified using a simple mathematical transformation. This transformation could be *hard-coded* into the synaptic connections between the retinal cells and the coincidence neurons. This method seems to be tedious and prone to error and it raises the question of how the human brain deals with rectification. Interestingly, it was shown by Benosman et al. (2011) that the epipolar geometry can be learned in an *unsupervised* fashion from interocular temporal correlations of two event-based visual streams. This idea can be directly applied to the stereo network. It not only provides a practical solution to the engineering problem, but it also makes a hypothesis about the way our brain deals with epipolar constraints in stereopsis. This is also discussed in further detail in the last chapter.

6.5.3 The problem of scaling

The neurons employed in the stereo network encode *absolute* disparities. For each spatial position in the three-dimensional disparity space of the field of view, one disparity neuron and two coincidence detectors are required. If the stereo network is to be implemented on a neuromorphic processor, this inevitably leads to a scaling problem because a three-dimensional structure has to be mapped onto a two-dimensional substrate. As a result, it has been demonstrated that emulating the stereo network with neuromorphic hardware is limited to very small scales. A possible solution to this problem would involve reducing the number of neurons with distinct disparity. Only a few layers of neurons with a preferred disparity close to zero could exist, which would restrict the perception of disparity to a small range around the fixation point of the sensors. The scene beyond this range could be observed by changing the fixation point, which would correspond to the mechanism of *vergence eye movement* in the human visual system. A few broadly tuned neurons (near and far cells) with smaller and larger disparities than the one defined by the observable disparity range could drive vergence. A method to efficiently implement vergence on a neuromorphic stereo vision system is further discussed in the final chapter.

6.5.4 Applicability

The applicability of stereo vision is very broad and the event-based approach is not more efficient in general. Whether a frame-based or event-based approach is more efficient depends on the requirements and constraints of the application. Indeed, neuromorphic vision is often very power efficient, has low latency and high dynamic range. However, a classical high-speed camera also has low latency, and a HDR camera has a wide dynamic range. This means that

if the only requirements for a certain application are latency or dynamic range, one of the well-grounded, frame-based techniques will likely provide a preferable approach. However, if both requirements are important, neuromorphic systems provide an excellent trade-off. This becomes immediately evident when looking at the example of automotive applications. State-of-the-art driving assistance systems employ front-faced cameras that continuously monitor the street. Such cameras can detect unpredictable events such as a car that suddenly breaks or an incautious pedestrian who walks onto the street. Obviously, these systems need to work reliably in dark and bright light conditions, which imposes significant requirements on the dynamic range of the cameras. To deal with this, multiple frames captured with distinct exposure times are combined to form a HDR image that is subsequently processed. This accumulation results in a significant latency, ranging from tens of milliseconds, up to one hundred milliseconds. Presuming the car moves at 120 Km/h, a 50 ms latency corresponds to a distance of 1.7 meters. In contrast, the latency of a neuromorphic system with a comparable dynamic range could be as low as a few milliseconds, which would correspond to a distance of only a few centimeters. Thus, this difference could easily be a matter of life or death in certain situations. Other applications involve different trade-offs. In this regard, a good trade-off between latency and power consumption could be decisive for augmented reality (AR) on future mobile devices (e.g. smart glasses).

7 Conclusion and Outlook

This chapter draws some conclusions about the work described in this thesis and suggests how it should continue. The findings and conclusions drawn from this work lead to the proposal of a novel event-based neuromorphic stereo vision system. This system is elaborated in detail and illustrates the potential of event-based neuromorphic systems and the impact they will have on the field of machine vision.

7.1 Summary and conclusion

The emergence of neuromorphic vision sensors launched the field of event-based machine vision. In turn, this has catalyzed a paradigm shift towards artificial vision systems inspired by the self-timed and data-driven computation of the brain. Based on the asynchronous sampling strategy employed by event-based cameras, this thesis proposes an innovative way of representing visual information in *space-time*, termed "time surfaces". Extracting spatio-temporal information from time surfaces has proven to be beneficial for solving the stereo correspondence problem. A novel event-based algorithm was proposed that carries out stereo matching based on the correlation of spatio-temporal features. As a result, this approach naturally exploits the motion present in the visual scene to enhance matching performance. The algorithm was shown to robustly solve the correspondence problem and extract precise depth information from various dynamic scenes. Although the algorithm adheres to the concept of event-based computation, it was found to be inefficient when trivially implemented, due to redundant computation. This is a general problem for many event-based machine vision algorithms. These algorithms tend not to be practical until they leverage the full potential of event-based computation. In order to do this, they need to be formulated in such a way that redundant computation is eliminated.

These efficiency considerations led to the idea of formulating a stereo algorithm in the form of a neural network. By revisiting historical work that describes cooperative networks for finding stereo correspondences, the foundation was laid for the proposed *spiking neural network* for stereo vision. It was found that by combining two simple mechanisms, *coincidence detection*

and *disparity evidence integration*, the stereo correspondence problem can be efficiently solved. These mechanisms were realized using *spiking, integrate-and-fire neurons*. The neural network uses the direct output of two event-based cameras, whereby events are handled as input spikes to the network. The neural network was shown to robustly respond to this input, precisely and correctly encoding disparity information. Thus, it has been demonstrated that the network is capable of solving the stereo correspondence problem. Despite its simplicity, the network is based on a model that makes use of key principles which are firmly rooted in both established and contemporary research in the field of stereopsis. In particular, the model suggests a simple and robust mechanism to explain how motion cues are integrated to help solve the correspondence problem. This mechanism is aligned with theories of disparity interactions and the computation of motion in depth. The model involves disparity-tuned neurons which have the ability to distinguish between true and false matches. This confirms recent neurophysiological findings (Read and Cumming, 2007; Haefner and Cumming, 2008) which suggest that some of the well-grounded models of stereopsis need to be revised. The model also makes predictions about the neural correlates of stereo perception in the cortex. Although it is commonly assumed that the extrastriate cortex plays a crucial role in solving the stereo correspondence problem, the proposed model suggests that it can be solved with simple mechanisms which are merely reliant on two types of cell found in the striate cortex. Yet another interesting aspect of the proposed model is its natural dependence on temporal dynamics. In the context of the field of psychophysics, for example, this could help to explain why motion-induced illusions, such as the Pulfrich effect, occur. Such illusions are still the subject of contradictory explanations (Read and Cumming, 2005). Finally, the proposed stereo network provides a model that fully leverages the benefits of event-based computation and thus, is optimally suited to being implemented on neuromorphic hardware.

Current machine vision processing systems face severe limitations due to the nature of the front-ends of conventional sensors and the classical Von Neumann computing architecture. Such sensors currently produce very large amounts of redundant data, because sequences of frames are sampled at fixed rates. In addition, the classical Von Neumann computing architecture is subject to the memory bottleneck (Backus, 1978) and requires high power and bandwidths to process continuous streams of images. The emerging field of neuromorphic engineering has produced efficient event-based sensors, that can produce low-bandwidth data in real time. In addition, powerful parallel computing architectures have emerged, which feature co-localized memory and computation and can carry out low-latency, event-based processing. This technology promises to solve many of the problems associated with conventional technologies in the field of machine vision. However, so far the progress has been chiefly technological, whereas related development of event-based models and signal processing algorithms has been comparatively lacking (with a few notable exceptions, e.g. see Section 3.5). This thesis elaborates an innovative model that can fully exploit the features of event-based visual sensors. In addtion, the model can be directly mapped onto existing neuromorphic processing architectures. A proof of concept demonstrates various ways in which the proposed model could be implemented in event-based, neuromorphic stereo vision

systems. The results show that the full potential is leveraged when single neurons from the stereo network are individually emulated in parallel. In order to emulate the full-scale stereo network, however, efficient neuromorphic hardware capable of emulating large-scale neural networks is required. Although a few promising approaches already exist, large-scale systems remain a challenge in neuromorphic engineering. A survey of potential applications of event-based, stereo vision systems revealed that there are various other uses for the technology that either do not require a full-scale stereo network, or that can be simulated rather than emulated. In particular, this is often the case if the power consumption requirements are not critical. Thus, one way or another, event-based stereo vision systems certainly have great potential.

7.2 Outlook

The results of this thesis have led to an innovative model for event-based stereo vision that has been shown to be ideally suited to implementation on neuromorphic hardware. Apart from the proof of concept, a full-scale, event-based neuromorphic stereo vision system that outperforms existing machine vision approaches has not been achieved. However, the research has advanced far enough to lay all of the groundwork for such a system. Indeed, in the findings and conclusion chapter of this thesis, a detailed architecture is proposed. This is illustrated in the form of a schematic of a neuromorphic, multi-core processor, explained in Section 7.2.1.

Although the obvious next step would be to develop the stereo chip, quite a few interesting research questions came up during this project. Basically, future research should focus on three different challenges. Firstly, the event-based algorithm and model need to be further refined. Secondly, a more biologically realistic model needs to be developed and validated on the basis of neurophysiological responses or phenomena. Finally, neuromorphic circuits capable of efficiently implementing the model need to be developed. These challenges are elaborated in more detail in the following paragraphs.

It has been shown that the proposed event-based STC algorithm is inefficient when implemented in its current form because it carries out redundant computation. An advanced implementation could be explored that avoids this redundancy. Here, the inspiration might stem from a number of existing approaches to machine vision that deal with this problem. A possible starting point might be the concept of *sliding windows* or *box filters*. Another similar concept, the *delta-ring buffer* proposed by Brändli (2015), describes a generalized method to formulate event-based algorithms which avoid redundant computation. As the STC algorithm is highly parallelizable, it would be interesting to investigate the best performance that could be achieved on a custom GPU. In the case of most of the simulations of the stereo network presented in this thesis, the model employed two equally sized populations of neurons, which served as the coincidence and disparity detectors respectively. It was shown that the disparity neurons are position invariant and thus are selective to the preferred disparities at any cyclopean position within their receptive field. This suggests that the model could be extended with more broadly spaced disparity neurons. In addition, different weighting functions for

retinal inputs could be studied, although the uniform receptive fields have proven to be very useful for implementation on hardware. Stereo rectification of the inputs was assumed to be a prerequisite for the stereo network model. It was proposed that unsupervised learning of the epipolar geometry, similar to research described in Benosman et al. (2011), could naturally arise due to interocular temporal coincidences. Such unsupervised learning could be directly incorporated into the synaptic connections that link the retinal cells and the coincidence detectors. This could be a very important feature of the model, rendering the laborious and delicate process of calibrating the sensors unnecessary.

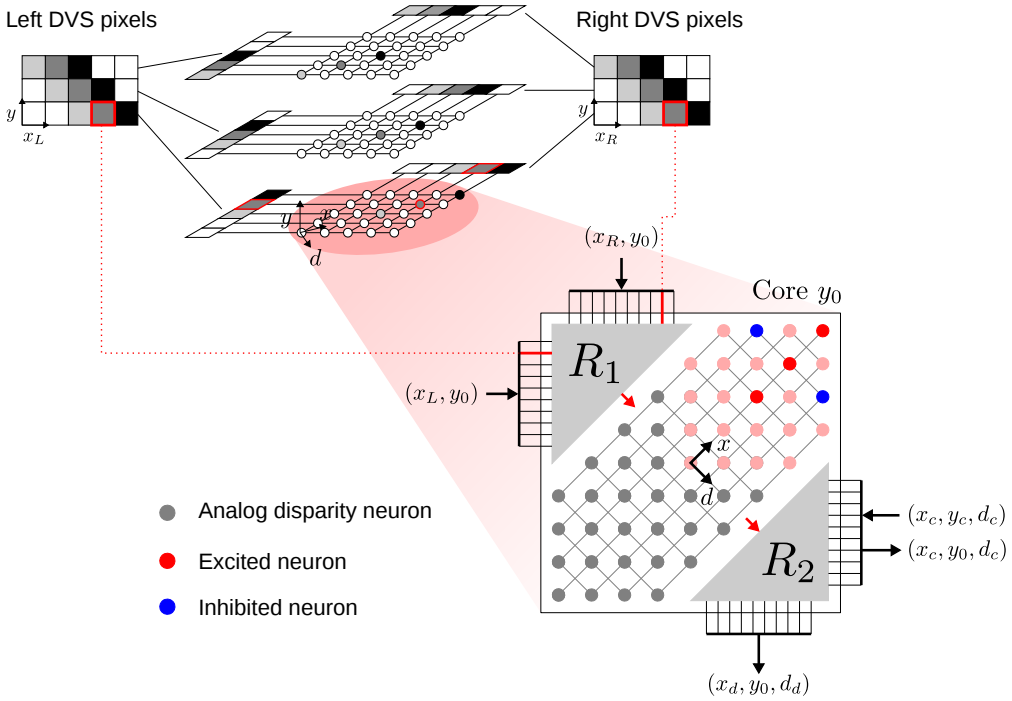
The interesting temporal properties of the proposed model informed the decision to link it to established models of stereopsis in the brain. Surprisingly, the majority of the existing stereopsis literature does not consider temporal dynamics. Although the model in its current form employs neurons whose behavior is highly simplified compared to real cells in the visual cortex, an extended version could incorporate established mechanisms such as spatial frequency and orientation tuning. In this sense, a single spike would no longer simply represent a change in temporal contrast. Instead, it would reflect a discrete component of a realistic neurophysiological response from a monocular cell. This continuative research chiefly aims to understand the process of stereopsis in the brain, rather than attempting to provide a model that can be efficiently implemented in an artificial system. Following a similar direction, further research could focus more on neurophysiological phenomena and investigate whether the model can explain some of the partially unexplained illusions. In this context, illusions that involve temporal aspects would be of particular interest, such as the Pulfrich and Flash-lag effects, or the Ternus illusion to name just a few. Furthermore, the bias of the model towards fronto-parallel motion suggests that illusions may appear if stimuli move both perpendicular and parallel to the visual axis simultaneously (see the “moving” double-nail experiment proposed in Chapter 5.4.5). If it were proved that this illusion exists, this would validate the present model and thus, it would have a major impact on the understanding of the role of temporal dynamics in the visual processing of the brain. Finally, the idea of unsupervised learning of epipolar geometry has raised the question of how the brain deals with changing epipolar constraints, as occur in the presence of vergence eye movements. One possibility would be that the brain employs many synapses, which cover the full range of possible epipolar constraints, and feature the effects of short-term plasticity or modification of synaptic transmission. These synapses would be driven by a signal which would encode the vergence eye position and thus enable and disable constraints accordingly. In this context, the spherical shape of the retina could play a crucial role in minimizing the degree of variability of the epipolar constraints and thus, the amount of synapses required. Of course, this is pure speculation but it gives an idea of where future research building upon the models and ideas in this thesis could lead.

Last but not least, a somewhat more concrete research proposition would be to further develop the neuromorphic circuits required to implement the proposed model. The obvious challenge is to build a large-scale system. A detailed proposal to achieve this aim is presented in the following subsection. In addition, neuromorphic circuits capable of efficiently and reliably

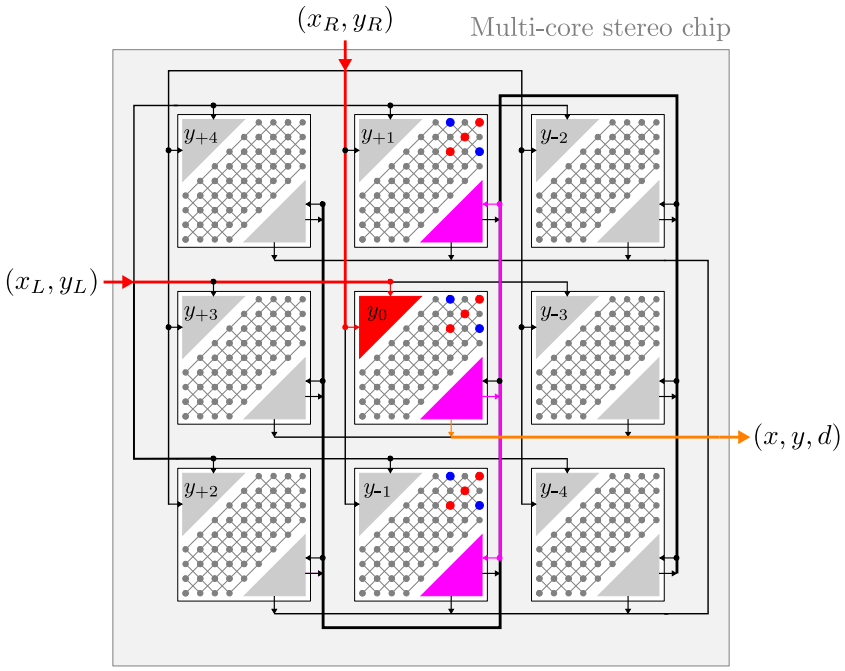
carrying out coincidence detection are needed. This would probably be best achieved using a digital approach. As a starting point, a simple AND gate could receive extended pulses as an input from two spiking sources, among which coincidences could be detected. Here, the width of the extended pulse would directly control the sensitivity to the interocular temporal delays of the input spikes. Alternatively, analog circuits could also provide a convenient way to implement coincidence detection. The previously discussed ROLLS processor employs an NMDA synapse which enables a voltage gating mechanism. When used together with a normal synapse, this mechanism could be exploited to achieve robust coincidence detection which would be less sensitive to transistor mismatch.

7.2.1 Towards a neuromorphic, multi-core stereo processor

The following subsection outlines a concrete proposal to implement the proposed large-scale stereo model using a neuromorphic, multi-core processor. The key principles employed by the processor are digital coincidence detection, analog disparity evidence integration and efficient asynchronous event routing. Fig. 7.1a illustrates the stereo network model, as detailed in Chapter 5. The stereo network employs neurons that homogeneously encode three-dimensional disparity space such that each neuron represents a unique spatial position $(x, y, d) \in \mathbb{D}^3$. Each horizontal plane with constant vertical cyclopean position y is implemented using a distinct core of the stereo processor. A particular core receives retinal events from one-dimensional arrays of pixels from both sensors, reflecting corresponding rectified epipolar lines, characterized by their common vertical coordinate y. Each core contains a two-dimensional array of disparity detectors that populate the space (x, d) at a fixed vertical position y. The disparity detectors are implemented as analog LIF silicon neurons. For this purpose, it is beneficial to use *analog silicon neuron* circuits. Such circuits can be very compact and their response is only marginally affected by mismatch because of the smoothing effect that naturally occurs due to the integration of many inputs. The disparity which can be encoded by the neurons is limited within a certain range, which is equally spread from the fixation point located at zero disparity, in order to clear a space for two digital routing fabrics, labeled R_1 and R_2 accordingly. The task of R_1 is to receive input events and determine temporal coincidences among pairs of interocular events. One way of achieving this might involve arrays of pulse extenders for each horizontal retinal position x_R and x_L respectively. If two pulses are active at the same time, a coincidence event is generated. The coincidence event is then internally routed to the relevant disparity neurons, as envisaged in the stereo model. Effectively, the targeted disparity neurons form a cross, centered at the origin (x_R, x_L) of the coincidence event, whereby the neurons on the diagonal ($d = x_R - x_L = const.$) are excited whereas those on the off-diagonal ($x = x_R + x_L = const.$) are inhibited as shown in the figure. The range of affected disparity neurons is determined by the width of the receptive field used in the model and could be a programmable parameter of R_1. In accordance with the stereo model, coincidence events are also integrated by the neighboring neurons at different vertical positions y. Therefore, the second router R_2 sends coincidence events to a bus that is shared among all the cores. Accordingly, the router R_2 of the neighboring cores receives coincidence events and inter-



(a)



(b)

Figure 7.1: A neuromorphic, multi-core stereo processor. **(a)** Architecture of a single core of the stereo processor. The core implements a horizontal layer of the stereo network, involving analog silicon neurons as disparity detectors and digital routing fabrics (R_1 and R_2) that implement coincidence detection and handle communication. **(b)** Layout of the multi-core stereo processor. The paths of retinal input events (red), inter-core coincidence events (purple) and disparity output events (orange) are highlighted accordingly.

nally distributes them among the disparity detectors in the same way described before. This scheme is illustrated in Fig. 7.1b for an example of a stereo processor with nine cores. In this example, the central core y_0 forwards a coincidence event to two neighboring cores, y_{-1} and y_{+1} respectively. The advantage of this scheme is that it minimizes inter-core communication. In accordance with the stereo model, a coincidence event is integrated by $2 \times \omega \times \omega$ disparity detectors. Whereas in the worst case, a different multi-core architecture would have to route $2 \times \omega \times \omega$ events for each coincidence event, the proposed scheme reduces the complexity to ω. Finally, if a disparity neuron spikes, a disparity event is written by R_2 onto a separate bus that provides the output of the chip. In this context, it should be considered that the stereo model requires that the disparity event should only be triggered if it was preceded by a coincidence event at the same disparity. However, since this information is locally present (within the same core) it should not be a problem to fulfill this requirement.

The proposed architecture can be further developed in various directions. If the capacity of the stereo processor is not sufficient, the interface could be extended such that it allows arbitrary scalability. This could be achieved by interconnecting as many stereo processors as required. Another possibility would be to use the concept of multiplexing, which would involve processing one sub-field of the overall visual field of the sensors at a time. This could be particularly interesting in the context of event-based systems in which the selection is driven by the activity. If these sub-fields would independently change their relative offsets, the system would perform a form of *virtual vergence*. This would enable the range of perceived depth to be extended in a similar way to that of the human visual system. Finally, a very interesting result could be obtained by joining the sensors and the stereo processor within a single system-on-chip (SoC). This approach would inevitably require sophisticated optics (such as mirrors or a biprism) to obtain two views with sufficient baseline. In terms of power efficiency, however, such an approach could outperform everything else that has been proposed so far.

7.2.2 Further future prospects

One of the major benefits of using event-based cameras for stereo vision is that the temporal matching criterion substantially simplifies the correspondence problem. In the context of stereo vision, however, a temporal matching criterion together with an epipolar constraint is not sufficient to solve the correspondence problem for complex scenes. Nevertheless, the use of additional cameras adds further epipolar constraints which enable simple temporal matching without sophisticated algorithms, even for complex scenes. An N-ocular 3D reconstruction algorithm for event-based vision has been proposed by Carneiro et al. (2013). However, the laborious calibration and the use of many sensors makes the system impractical and expensive. Future research in this direction could employ a population of coincidence detectors, as described in the proposed stereo model, which would detect coinciding events from multiple rather than two views. Due to the presence of multiple epipolar constraints, coincidence detection alone could be sufficient, without requiring the costly process of integrating disparity evidence. The coincidence detection mechanism could be relatively cheaply

implemented in silicon, based on the ideas described in the previous section. The complex epipolar constraints could be enforced by programmable synapses, which would link the retinal pixels with the network of coincidence detectors. In this context too, these constraints could be automatically learned from temporal coincidences, similarly to the method previously described. This system would have the lowest possible latency (in the range of microseconds) because almost no processing apart from coincidence detection is carried out. Theoretically, this simple algorithm could even be integrated as a coprocessor together with a single sensor chip, whereby the pixels would be divided into equally sized areas, each of which would represent a different view, observed through a light-field lens. This idea would not only tackle the problem of fragile calibration but it could lead to an innovative, ultra-fast, cheap and most importantly passive, 3D sensor.

A Appendix

A.1 Epipolar geometry

A.1.1 Direct linear transformation (DLT) algorithm

The direct linear transformation (DLT) algorithm is explained here for reference. For further explanation, particularly concerning mathematical details, the reader is referred to the original literature (Hartley and Zisserman, 2004). In its simplest form, the DLT method describes a *linear* algorithm for determining the homography H between 2D-2D point correspondences, i.e. $\mathbf{x}'_i = H\mathbf{x}_i$. Note that this transformation involves homogeneous coordinates. Thus, the equation is defined up to scale. When expressed in a vector cross product, as $\mathbf{x}'_i \times H\mathbf{x}_i = 0$, a simple linear solution may be derived:

$$\mathbf{x}'_i \times H\mathbf{x}_i = \begin{pmatrix} y'_i\mathbf{h}^{3\top}\mathbf{x}_i - w'_i\mathbf{h}^{2\top}\mathbf{x}_i \\ w'_i\mathbf{h}^{1\top}\mathbf{x}_i - x'_i\mathbf{h}^{3\top}\mathbf{x}_i \\ x'_i\mathbf{h}^{2\top}\mathbf{x}_i - y'_i\mathbf{h}^{1\top}\mathbf{x}_i \end{pmatrix} \tag{A.1}$$

where $\mathbf{h}^{j\top}$ denotes the j-th row of the matrix H and $\mathbf{x}^i_i = \begin{bmatrix} x'_i & y'_i & w'_i \end{bmatrix}^\top$. Since $\mathbf{h}^{j\top}\mathbf{x}_i = \mathbf{x}_i^\top\mathbf{h}^j$ Eq. A.1 can be rewritten in the form:

$$\begin{bmatrix} \mathbf{0}^\top & -w'_i\mathbf{x}_i^\top & y'_i\mathbf{x}_i^\top \\ w'_i\mathbf{x}_i^\top & \mathbf{0}^\top & -x'_i\mathbf{x}_i^\top \\ -y'_i\mathbf{x}_i^\top & x'_i\mathbf{X}_i^\top & \mathbf{0}^\top \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = 0 \tag{A.2}$$

This set of equations has the form $A_i\mathbf{h} = 0$, whereby A_i is a 3×9 matrix and $\mathbf{h}$ is a row vector

derived from the entries of H:

$$\mathbf{h} = \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix}, \qquad H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \tag{A.3}$$

Each point correspondence yields a set of three equations as described in Eq. A.2, of which only two are linearly independent. In order to solve for H, these two equations may be stacked for all the point correspondences yielding the linear set of equations $A\mathbf{h} = 0$, which can then be easily solved. Since H is only defined up to scale, four point correspondences are sufficient to find a solution for h. In this case, A is a 8×9 matrix with rank 8, thus providing a solution that is determined up to scale. The scale can be chosen arbitrarily, e.g. $||\mathbf{h}|| = 1$.

A.1.2 Linear triangulation

The linear triangulation method is based on the DLT algorithm (see Appendix A.1.1) and also described in detail in Hartley and Zisserman (2004). Given two matched image points $\mathbf{x}$ and $\mathbf{x}'$ with associated camera matrices P and P', the image points are projections of a common 3D point $\mathbf{X}$, i.e. $\mathbf{x} = P\mathbf{X}$ and $\mathbf{x}' = P'\mathbf{X}$. A set of three equations, of which two are linearly independent, can be obtained from each image point by computing the vector cross product $\mathbf{x} \times (P\mathbf{X}) = 0$:

$$\begin{aligned} x\left(\mathbf{p}^{3\top}\mathbf{X}\right) - \left(\mathbf{p}^{1\top}\mathbf{X}\right) &= 0 \\ y\left(\mathbf{p}^{3\top}\mathbf{X}\right) - \left(\mathbf{p}^{2\top}\mathbf{X}\right) &= 0 \\ x\left(\mathbf{p}^{2\top}\mathbf{X}\right) - y\left(\mathbf{p}^{1\top}\mathbf{X}\right) &= 0 \end{aligned} \tag{A.4}$$

where $\mathbf{p}^{i\top}$ is the i-th row of P. The first two equations from Eqs. A.4 can be stacked for both image points and a linear equation of the form $A\mathbf{X} = 0$ is obtained, whereby

$$A = \begin{bmatrix} x\mathbf{p}^{3\top} - \mathbf{p}^{1\top} \\ y\mathbf{p}^{3\top} - \mathbf{p}^{2\top} \\ x'\mathbf{p}'^{3\top} - \mathbf{p}'^{1\top} \\ y'\mathbf{p}'^{3\top} - \mathbf{p}'^{2\top} \end{bmatrix} \tag{A.5}$$

The homogeneous solution is found to be the smallest eigenvector of $A^\top A$ associated with the smallest eigenvalue or equivalently, the unit singular vector corresponding to the smallest singular value of A (see Hartley and Zisserman (2004) for more details).

A.2 Camera calibration

The simplest way to compute the camera projection matrix P is by using a DLT from a sufficient number of correspondences between a 3D point $\mathbf{X}$ and its image $\mathbf{x}$. This method is also described in detail in Hartley and Zisserman (2004) and summarized below.

Given a number of point correspondences between 3D points $\mathbf{X}_i$ and 2D image points $\mathbf{x}_i$, the goal is to find the 3×4 camera matrix P such that $\mathbf{x}_i = P\mathbf{X}_i$. Similarly to how it is described in Appendix A.1.1, the following relationship for each correspondence can be derived:

$$\begin{bmatrix} \mathbf{0}^\top & -w_i\mathbf{X}_i^\top & y_i\mathbf{X}_i^\top \\ w_i\mathbf{X}_i^\top & \mathbf{0}^\top & -x_i\mathbf{X}_i^\top \\ -y_i\mathbf{X}_i^\top & x_i\mathbf{X}_i^\top & \mathbf{0}^\top \end{bmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = 0 \tag{A.6}$$

where $\mathbf{p}^i$ is the i-th row of P. Since the three equations are linearly dependent, it is sufficient to use only the first two equations:

$$\begin{bmatrix} \mathbf{0}^\top & -w_i\mathbf{X}_i^\top & y_i\mathbf{X}_i^\top \\ w_i\mathbf{X}_i^\top & \mathbf{0}^\top & -x_i\mathbf{X}_i^\top \end{bmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = 0 \tag{A.7}$$

The projection matrix P has 12 entries but is defined up to scale. This means it has 11 degrees of freedom. At least 6 correspondences providing 12 equations are required (of which only 11 are needed) to determine the projection matrix P. By stacking up Eqs. A.7 from $n \geq 6$ correspondences, a $2n \times 12$ matrix A is obtained. From A, the projection matrix P can be computed by solving the set of equations $A\mathbf{p} = 0$, whereby $\mathbf{p}$ is the vector containing the elements of P. Usually, image points are noisy, which suggests that more than the minimum number of point correspondences should be used. The set of equations is then *over-determined* and the solution will be the smallest eigenvector of $A^\top A$ associated with the smallest eigenvalue or equivalently, the unit singular vector corresponding to the smallest singular value of A (see Hartley and Zisserman (2004) for more details).

The disadvantage of this simple calibration method is that in order to estimate the projection matrix accurately, a large set of correspondences between physical 3D points and 2D image points are required. In addition, the 3D points must not all lie on the same plane. In cases where such a set of correspondences is available, it often does not cover the entire field of view, which can lead to very inaccurate calibration results. A more flexible method, which has been widely adopted, is described in Zhang (1999). Here, the projection matrix can be computed from a few (at least two) different views of a planar pattern. For each such view, a number of 2D-2D correspondences between points of the pattern (expressed in the 2D coordinate

system of the pattern) and image points are required. However, neither the position, nor the orientation of the pattern itself, are necessary. Either the camera or the pattern can be moved to generate the different views. A summarized description of the method is described below, while the complete details can be found in Zhang (1999).

Given a projection matrix $P = K\begin{bmatrix} R & | & \mathbf{t} \end{bmatrix}$ with extrinsic parameters R and $\mathbf{t}$ and the intrinsic camera matrix

$$K = \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \quad , \tag{A.8}$$

the relationship between an image point $\mathbf{x}_i$ and a point on the model plane $\mathbf{X}_i$ (whereby without loss of generality the model plane is assumed at $Z = 0$) is

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K\begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = K\begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = H\tilde{\mathbf{x}}_i \tag{A.9}$$

where $\mathbf{r}_i$ is the i-th column of R, $\tilde{\mathbf{x}}_i$ is a model point and the 3×3 matrix $H = K\begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix}$ is the homography between the image and model plane. Since $\mathbf{r}_1$ and $\mathbf{r}_2$ are orthonomal, two basic constraints may be derived:

$$\begin{aligned} \mathbf{h}_1^\top K^{-\top} K^{-1} \mathbf{h}_2 &= 0 \\ \mathbf{h}_1^\top K^{-\top} K^{-1} \mathbf{h}_1 &= \mathbf{h}_2^\top K^{-\top} K^{-1} \mathbf{h}_2 \end{aligned} \tag{A.10}$$

where $\mathbf{h}_i$ is the i-th column of H. Given one homography, only two constraints on the intrinsic parameters can be derived, as a homography has eight degrees of freedom and there are six unknown extrinsic parameters. Now, consider the symmetric matrix

$$Q = K^{-\top} K^{-1} = \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \tag{A.11}$$

and let $\mathbf{q}$ be a 6-dimensional vector containing the entries from Q, i.e. $\mathbf{q} = \begin{bmatrix} q_{11} & q_{12} & q_{22} & q_{13} & q_{23} & q_{33} \end{bmatrix}^\top$.

If the i-th column vector of H is $\mathbf{h}_i = \begin{bmatrix} h_{i1} & h_{i2} & h_{i3} \end{bmatrix}$, it follows

$$\mathbf{h}_i^\top Q \mathbf{h}_j = \mathbf{v}_{ij}^\top \mathbf{q} \tag{A.12}$$

with $\mathbf{v}_{ij} = \begin{bmatrix} h_{i1}h_{j1} & h_{i1}h_{j2}+h_{i2}h_{j1} & h_{i2}h_{j2} & h_{i3}h_{j1}+h_{i1}h_{j3} & h_{i3}h_{j2}+h_{i2}h_{j3} & h_{i3}h_{j3} \end{bmatrix}^\top$. Eq. A.10 can then be rewritten in homogeneous form as follows:

$$\begin{bmatrix} \mathbf{v}_{12}^\top \\ (\mathbf{v}_{11}-\mathbf{v}_{22})^\top \end{bmatrix} \mathbf{q} = 0 \tag{A.13}$$

Each image yields two equations. The equations from n images can be stacked, which yields $V\mathbf{q} = 0$, where V is a $2n \times 6$ matrix. Thus, $n \geq 3$ is required for a general unique solution. If the skewness constraint $s = 0$ is enforced, two images ($n = 2$) are sufficient. In that case, an additional equation $\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^\top \mathbf{q} = 0$ is added. As before, the solution is known to be the smallest eigenvector of $V^\top V$ associated with the smallest eigenvalue or equivalently, the unit singular vector corresponding to the smallest singular value of V.

The two methods presented above both minimize an *algebraic* error. Often, this linear estimate is then used as a starting point for an iterative algorithm — for example the famous Levenberg-Marquardt method — to minimize a more significant *geometric* error. Furthermore, today's standard calibration routines also incorporate models for lens distortion, which is not discussed here but can be found in the relevant literature (Hartley and Zisserman, 2004; Zhang, 1999).

A.2.1 Event-based calibration

Two methods, which were developed during this project to calibrate event-based cameras, are presented: the *vibrating sensors* and the *flashing patterns* method.

Vibrating sensors method

Event-based cameras can be very simply and cheaply calibrated by being mounted on a vibrating substrate. This could be a simple tripod with a flexible component that can be manually stimulated so that it resonates for a few seconds. The magnitude of the vibration should be small enough that the projected calibration pattern only moves within the distance of one pixel. By accumulating the events over a few seconds, histogram images can be produced from which the calibration pattern can be detected. While this method is very practical and does not require sophisticated equipment, it is not expected to be very precise. This method was tested with a stereo rig involving two DVS cameras, each of which had a 128×128-pixel resolution. The calibration model consisted of two white orthogonal planes containing 12 black, 25 mm squares. Fig. A.1a shows the histograms obtained after events from both cameras

were accumulated for two seconds. Fig. A.1b shows the manually selected corners and their reprojection after calibration. The calibration method used corresponds to the first algorithm described in Appendix A.2 without the refinement procedure or consideration of lens distortion. The algebraic residual for the left camera was $||A\mathbf{p}|| = 0.036$ and for the right camera $||A\mathbf{p}|| = 0.054$. The mean reprojection errors in pixels are $e_{MRE} = 0.41$ for the left camera and $e_{MRE} = 0.47$ for the right camera.

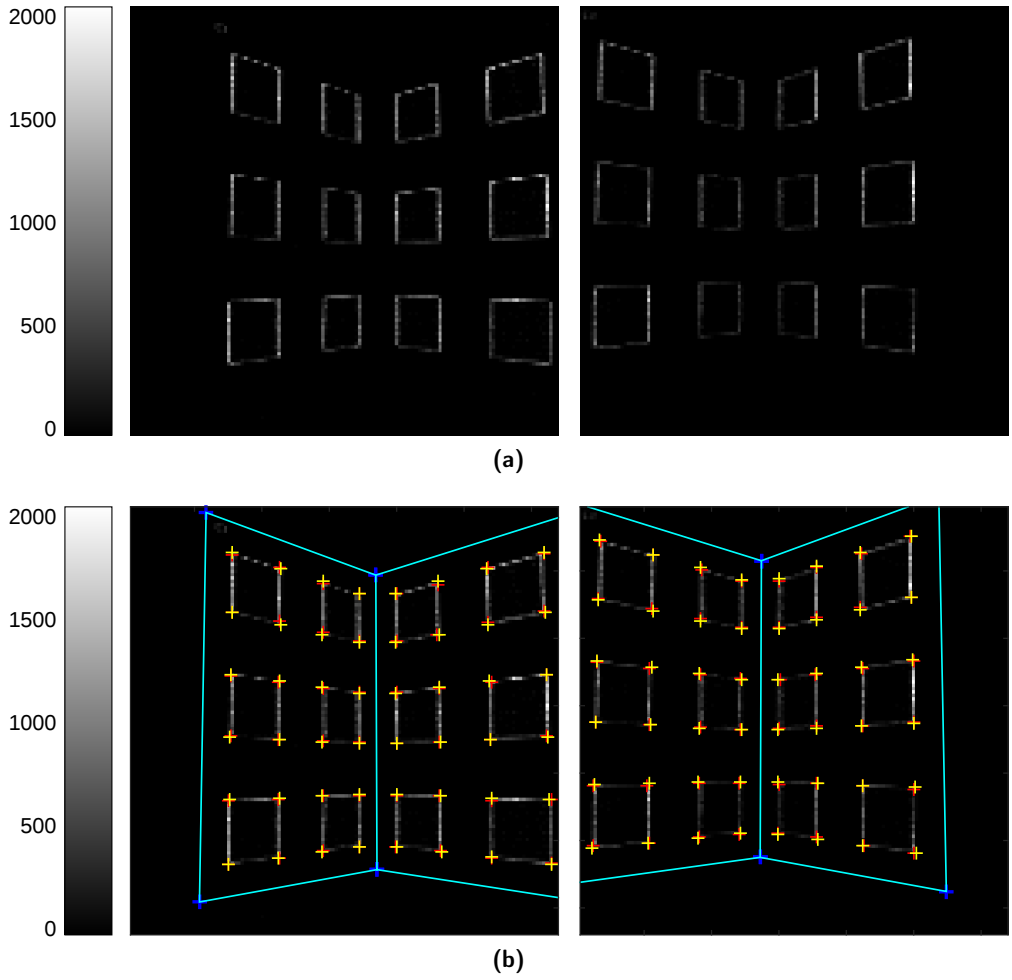


Figure A.1: Vibrating sensors calibration method. **(a)** Accumulated event histograms ($T = 2$ s) from two DVS128 sensors. **(b)** Visualization of the calibration results. Yellow crosses indicate positions of manually detected corners. Red crosses indicate the corresponding reprojected corners from the ground truth model of the calibration pattern using the projection matrix obtained from the calibration procedure (see Appendix A.2). The reprojection of the calibration model outline is shown in blue.

Projection matrices:

$$P_L = \begin{pmatrix} -0.0180 & 0.0010 & 0.0368 & -0.9617 \\ 0.0125 & -0.0368 & 0.0125 & -0.2676 \\ 0.0002 & 0.0000 & 0.0002 & -0.0132 \end{pmatrix}, \qquad P_R = \begin{pmatrix} 0.0233 & -0.0014 & -0.0496 & 0.9556 \\ -0.0164 & 0.0498 & -0.0153 & 0.2837 \\ -0.0002 & -0.0000 & -0.0002 & 0.0178 \end{pmatrix}$$

From the projection matrix $P = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \end{bmatrix}$, the intrinsic and extrinsic parameters can be derived with a QR decomposition of $M = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 \end{bmatrix} = K \cdot R$.

Intrinsic parameters:

$$K_L = \begin{pmatrix} 171.9739 & 0.3179 & 59.5003 \\ 0 & -168.4012 & 67.3152 \\ 0 & 0 & 1.0000 \end{pmatrix}, \qquad K_R = \begin{pmatrix} 176.5837 & -0.3503 & 53.8816 \\ 0 & -172.8690 & 62.8269 \\ 0 & 0 & 1.0000 \end{pmatrix}$$

Note that the rotation matrices were computed from the QR decomposition. Thus, they do not necessarily satisfy the properties of a rotation matrix:

$$R_L = \begin{pmatrix} 0.7078 & 0.0003 & -0.7064 \\ 0.0484 & -0.9977 & 0.0480 \\ -0.7047 & -0.0682 & -0.7062 \end{pmatrix}, \qquad R_R = \begin{pmatrix} 0.6710 & -0.0059 & -0.7415 \\ 0.0511 & -0.9972 & 0.0542 \\ -0.7397 & -0.0742 & -0.6688 \end{pmatrix}$$

The camera centers are derived from $\mathbf{c} = M^{-1}\mathbf{p}_4$:

$$\mathbf{c}_L = \begin{pmatrix} 37.3297 \\ 20.3391 \\ 43.7970 \end{pmatrix}, \qquad \mathbf{c}_R = \begin{pmatrix} 43.3984 \\ 20.5987 \\ 39.1166 \end{pmatrix}$$

Flashing patterns method

For the method described here, a normal computer monitor is used to display a flashing calibration pattern that is then captured by the event-based cameras. As before, calibration images are generated from the accumulation of events. The flashing pattern is generated such that the accumulated images can directly be processed by standard calibration toolboxes, such as the one included with OpenCV[1]. It was found that it is helpful to smooth the generated images before passing them to the calibration pipeline. An example of two calibration images is shown in Fig. A.2b. The main advantage of this method is that fully automated calibration toolboxes can be used, including *automatic pattern detection*, *undistortion* and *refinement*

[1] http://docs.opencv.org/doc/tutorials/calib3d/cameracalibration/cameracalibration.html

procedures. The OpenCV calibration toolbox used here is based on the second method described in Appendix A.2. The calibration results show a significantly smaller reprojection error $e_{MRE} = 0.249$ than was obtained using the *vibrating sensors* method. In total, $n = 16$ generated calibration images of a 7×5 checkerboard consisting of alternating black and white 50 mm squares were used.

Intrinsic parameters:

$$K_L = \begin{pmatrix} 153.55 & 0 & 55.58 \\ 0 & 153.48 & 74.54 \\ 0 & 0 & 1 \end{pmatrix}, \qquad K_R = \begin{pmatrix} 153.64 & 0 & 56.31 \\ 0 & 153.83 & 65.85 \\ 0 & 0 & 1 \end{pmatrix}$$

Extrinsic parameters:

$$R = \begin{pmatrix} 0.9940 & -0.0023 & 0.1085 \\ 0.0017 & 1 & 0.0056 \\ -0.1085 & -0.0054 & 0.9940 \end{pmatrix}, \qquad T = \begin{pmatrix} -8.805 \\ -0.078 \\ 0.636 \end{pmatrix}$$

Fundamental matrix (after undistortion):

$$F^\star = \begin{pmatrix} 1.0673 \cdot 10^{-6} & -9.1774 \cdot 10^{-5} & 0.0050 \\ -4.6590 \cdot 10^{-5} & -7.0253 \cdot 10^{-6} & 0.1983 \\ 0.0044 & -0.1897 & 1 \end{pmatrix}$$

Essential matrix (after undistortion):

$$E^\star = \begin{pmatrix} 0.0074 & -0.6357 & -0.0812 \\ -0.3233 & -0.0487 & 8.8222 \\ 0.0627 & -8.8054 & -0.0407 \end{pmatrix}$$

Fundamental matrix (8-Point algorithm, without undistortion):

$$F = \begin{pmatrix} 1.0258 \cdot 10^{-6} & -9.9164 \cdot 10^{-5} & 0.0058 \\ -3.2143 \cdot 10^{-5} & 2.1347 \cdot 10^{-5} & 0.1758 \\ 0.0034 & -0.1716 & 1 \end{pmatrix}$$

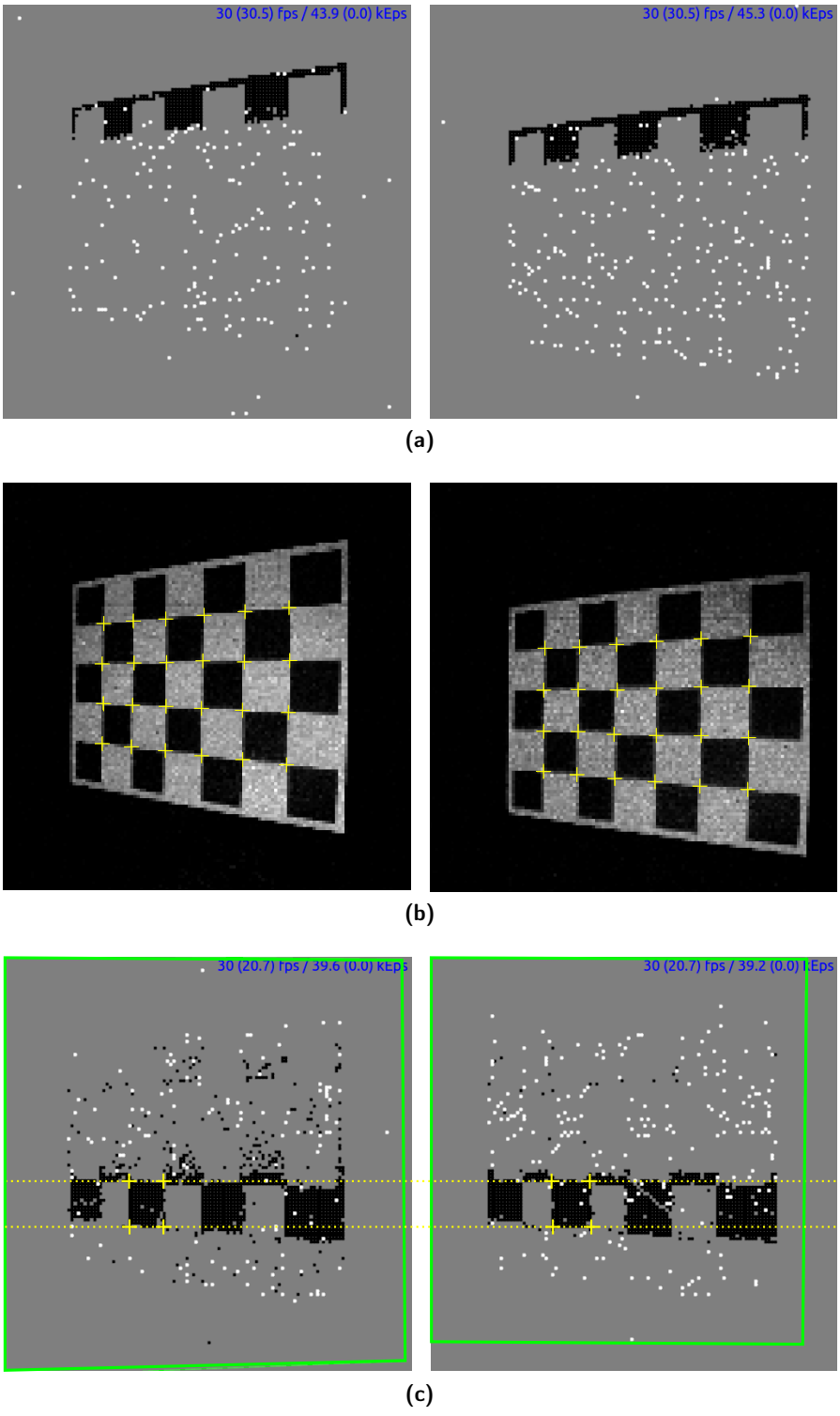


Figure A.2: Flashing patterns calibration method. **(a)** Snapshot of the live event stream rendered at 30 Hz, showing the top-to-bottom refresh mechanism of the LCD monitor. **(b)** Accumulated calibration images from the events of a flashing 7×5 checkerboard. The yellow crosses indicate automatically detected positions of the corners of the calibration pattern OpenCV. **(c)** Snapshot of the undistorted and rectified event stream using the obtained calibration data. Two random epipolar lines are visualized. Due to the rectification procedure, the epipolar lines are perfectly horizontally aligned.

Homographic transformation matrices for rectification (derived from F):

$$H_L = \begin{pmatrix} -0.1604 & -0.0021 & 0.9700 \\ 0.0033 & -0.1717 & 1.3235 \\ 3.1001 \cdot 10^{-5} & -2.1301 \cdot 10^{-5} & -0.1695 \end{pmatrix}, \quad H_R = \begin{pmatrix} 1.0359 & 0.0077 & -2.7887 \\ 0.02844 & 1.0002 & -1.8355 \\ 0.0006 & 0 & 0.9638 \end{pmatrix}$$

A.2.2 Frame-based calibration

The capability of the DAVIS sensor to read out conventional grayscale images means that existing camera calibration toolboxes can be directly used. Here, the results of the stereo calibration of two DAVIS20b sensors using Matlab's Stereo Camera Calibration App[2] are presented. In total, 19 calibration images were used and an average reprojection error of $e_{MRE} = 0.13$ was obtained. This suggests a high level of accuracy that significantly outperforms the event-based calibration methods.

Intrinsic parameters:

$$K_L = \begin{pmatrix} 251.0346 & 0 & 149.0265 \\ 0 & 250.5646 & 83.8300 \\ 0 & 0 & 1.0000 \end{pmatrix}, \quad K_R = \begin{pmatrix} 251.0120 & 0 & 138.7635 \\ 0 & 250.5476 & 88.9187 \\ 0 & 0 & 1.0000 \end{pmatrix}$$

Extrinsic parameters:

$$R = \begin{pmatrix} 0.9989 & 0.0094 & 0.0459 \\ -0.0085 & 0.9998 & -0.0180 \\ -0.0461 & 0.0176 & 0.9988 \end{pmatrix}, \quad T = \begin{pmatrix} -96.1629 \\ 0.8416 \\ 0.1186 \end{pmatrix}$$

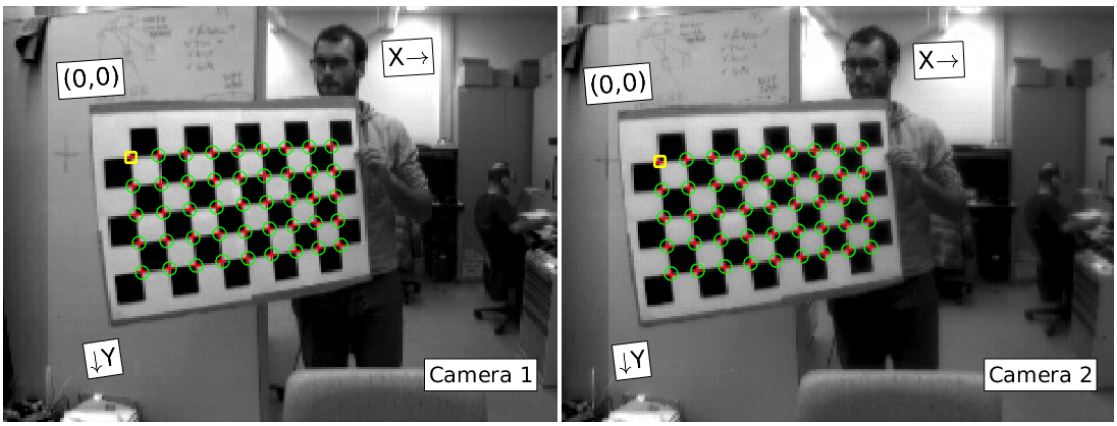
Fundamental matrix:

$$F = \begin{pmatrix} 0.0000 & 0.0001 & -0.0061 \\ 0.0000 & -0.0000 & 0.3859 \\ -0.0036 & -0.3913 & 1.4221 \end{pmatrix}$$

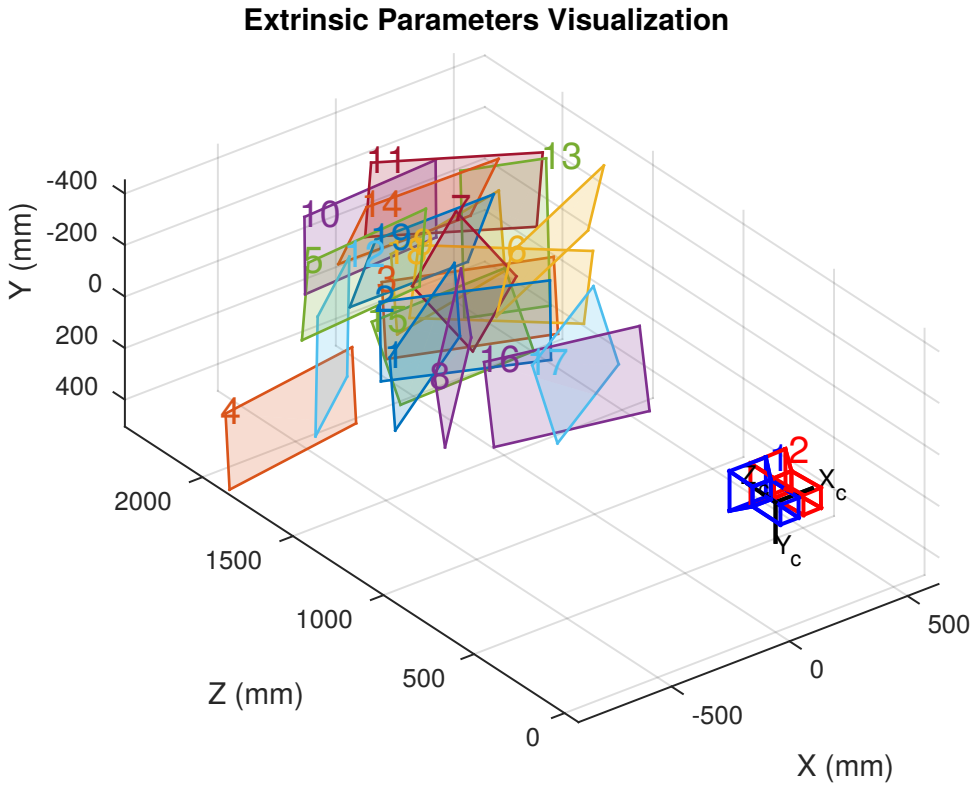
Essential matrix:

$$E = \begin{pmatrix} 0.0378 & 4.3138 & 0.0188 \\ 0.1038 & -1.6917 & 96.1517 \\ -0.8427 & -96.0512 & -1.6916 \end{pmatrix}$$

[2] http://www.mathworks.com/help/vision/ref/stereocameracalibrator-app.html



(a)



(b)

Figure A.3: Frame-based stereo camera calibration of two DAVIS sensors. **(a)** Screenshot of Matlab's calibration app, showing corresponding calibration images taken with two DAVIS240b sensors. The checkerboard is automatically detected. Green circles indicate detected corners and the red crosses their reprojections. **(b)** Visualization of the extrinsic parameters. The cameras and the poses of the 19 calibration patterns are shown in the world coordinate system.

Radial lens distortion:

$$k_L = \begin{pmatrix} -0.3256 & 0.1471 \end{pmatrix}, \qquad k_R = \begin{pmatrix} -0.3242 & 0.1445 \end{pmatrix}$$

A.3 Event-based visual flow

The mathematical definition of visual flow as described in this section does not correspond to the real *optical flow* as defined in machine vision. In fact, there are quite a few problems with the definition described here. Therefore, it should not be considered as a general equation of visual flow. Instead, it serves as an experimental test bed that provides some useful insights despite its shortcomings.

A.3.1 Derivation of visual flow from time surfaces

The visual flow, as defined in Benosman et al. (2014), can be directly derived from the temporal image Σ_e or the time surfaces Λ or Γ. Here, it is demonstrated that it can be derived from the exponential time surface (ETS), denoted as Γ. In this light, consider the following partial derivatives

$$\begin{array}{rcccl} \dfrac{\partial \Gamma(x,y,t)}{\partial x} & = & \dfrac{d\Gamma(x,y,t)}{dt}\dfrac{dt}{dx} & = & \dfrac{d\Gamma(x,y,t)}{dt}\dfrac{1}{v_x} \\ \dfrac{\partial \Gamma(x,y,t)}{\partial y} & = & \dfrac{d\Gamma(x,y,t)}{dt}\dfrac{dt}{dy} & = & \dfrac{d\Gamma(x,y,t)}{dt}\dfrac{1}{v_y} \end{array} \tag{A.14}$$

where v_x and v_y are the components of the estimated pixel velocity $\mathbf{v} = (v_x.v_y)^\top$. The velocity at spatial location (x_0, y_0) at time t can then be expressed as:

$$\mathbf{v}(x_0,y_0,t) = -\frac{1}{\tau}\Gamma(x_0,y_0,t)\left(\left(\frac{\partial \Gamma(x,y_0,t)}{\partial x}\Big|_{x=x_0}\right)^{-1}, \left(\frac{\partial \Gamma(x_0,y,t)}{\partial y}\Big|_{y=y_0}\right)^{-1}\right)^\top$$

The partial derivatives can be expressed in the shortened forms $\Gamma_x = \frac{\partial \Gamma(x,y,t)}{\partial x}$ and $\Gamma_y = \frac{\partial \Gamma(x,y,t)}{\partial y}$ respectively. For an event $e(\mathbf{p_0}, s, t)$, the surface at $\Gamma(x_0, y_0, t)$ is equal to the event's polarity s. As a result, the term can be simplified and referred to as the *velocity of an event*:

$$\mathbf{v}(e) = -\frac{s}{\tau}\left(\frac{1}{\Gamma_x(\mathbf{p_0},t)}, \frac{1}{\Gamma_y(\mathbf{p_0},t)}\right)^\top \tag{A.15}$$

Note that, as it stands, this definition of $\mathbf{v}$ is problematic as the derivation of the time surface along a *spatial edge* yields a component that is close to zero, which, in turn, corresponds

to a velocity component that approaches infinity. This is a problem with the current definition which is related to the *correspondence problem* (because the velocity was derived from physically non-corresponding events) and the *aperture problem* (the velocity component in the direction of the orientation of a spatial edge cannot be locally determined). To obtain a more meaningful and practical velocity for the purposes of this research, it is proposed to consider the projection of $\mathbf{v}$ onto the *normal* of the spatial edge (which is perpendicular to the orientation of the edge).

A.3.2 Cross-correlating visual flow

Consider a single point moving in 3D space with a velocity of $\mathbf{v}(t)$. For the sake of simplicity, assume that in this case, the point is projected to single events e_L and e_R of the same polarity $s_L = s_R = s$ in both views. It is also assumed that the projected velocities of those events have the same direction, meaning that

$$\frac{\mathbf{v}(e_L)}{|\mathbf{v}(e_L)|} = \frac{\mathbf{v}(e_R)}{|\mathbf{v}(e_R)|} = \mathbf{u} \tag{A.16}$$

The image coordinate system is substituted with $\mathbf{e}_u = \mathbf{u}$ and $\mathbf{e}_w = \mathbf{w}$, whereby $\mathbf{w} \perp \mathbf{u}$ and the new origin are chosen at the location of the event $\mathbf{p_0}$. The projected velocities of the events are obtained from Eq. A.15 and can be simplified to scalars as follows:

$$\begin{aligned} v_L &= \frac{1}{\tau \Gamma_u^L(0,0,t)} \\ v_R &= \frac{1}{\tau \Gamma_u^R(0,0,t)} \end{aligned} \tag{A.17}$$

Now consider the spatiotemporal features as defined in Eq. 4.8. The features are exponentials when located on the negative u-axis and zero everywhere else:

$$\begin{aligned} F_e^L(u,w) &= \mathrm{e}^{\frac{u}{\tau_L}} \sigma(-u)\delta(w) \\ F_e^R(u,w) &= \mathrm{e}^{\frac{u}{\tau_R}} \sigma(-u)\delta(w) \end{aligned} \tag{A.18}$$

where $\sigma(\cdot)$ is the step and $\delta(\cdot)$ the dirac function and τ_L and τ_R are the decay times that are given by

$$\begin{aligned} \tau_L &= \frac{1}{\Gamma_u^L(0,0,t)} &= v_L \tau \\ \tau_R &= \frac{1}{\Gamma_u^R(0,0,t)} &= v_R \tau \end{aligned} \tag{A.19}$$

Placing Eq. A.19 in Eq. A.18, the correlation of the features as defined in Eq. 4.11 can now be

calculated, assuming infinitely small pixels and an infinitely large neighborhood, such that boundary effects can be neglected:

$$\rho(e_L, e_R) = \frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(F_e^L - \mu_L)(F_e^R - \mu_R)\mathrm{d}u\mathrm{d}w}{\sqrt{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(F_e^L - \mu_L)^2\mathrm{d}u\mathrm{d}w \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(F_e^R - \mu_R)^2\mathrm{d}u\mathrm{d}w}}$$

where $\mu_L = \mu_R = 0$ because $\lim_{n\to\infty}\left(\frac{1}{n}\int_{-n}^{0}\mathrm{e}^u\mathrm{d}u\right) = 0$. Thus,

$$\begin{aligned}
\rho(e_L, e_R) &= \frac{\int_{-\infty}^{\infty}\int_{-\infty}^{0}\mathrm{e}^{u/v_L\tau}\mathrm{e}^{u/v_R\tau}\delta(w)\mathrm{d}u\mathrm{d}w}{\sqrt{\int_{-\infty}^{\infty}\int_{-\infty}^{0}\mathrm{e}^{2u/v_L\tau}\delta(w)\mathrm{d}u\mathrm{d}w \int_{-\infty}^{\infty}\int_{-\infty}^{0}\mathrm{e}^{2u/v_R\tau}\delta(w)\mathrm{d}u\mathrm{d}w}} \\
&= \frac{\int_{-\infty}^{0}\mathrm{e}^{u/\tau(1/v_T+1/v_R)}\mathrm{d}u}{\sqrt{\int_{-\infty}^{0}\mathrm{e}^{2u/v_L\tau}\mathrm{d}u\int_{-\infty}^{0}\mathrm{e}^{2u/v_R\tau}\mathrm{d}u}} \\
&= \frac{\tau\left(\frac{1}{v_L}+\frac{1}{v_R}\right)^{-1}}{\sqrt{\frac{\tau^2 v_L v_R}{4}}}
\end{aligned}$$

From here, the final formula can be derived as follows:

$$\rho(e_L, e_R) = \rho(v_L, v_R) = 2\frac{\sqrt{v_L v_R}}{v_L + v_R} \tag{A.20}$$

It can be easily shown that this equation not only holds for single events, but also for multiple events, that move together and thus form a spatial pattern. However, this is only true if the spatial pattern is arranged such that the traces of the events do not overlap. Furthermore, it is assumed that within a feature, all of the events move at the same velocity v_L and v_R respectively. When these assumptions hold, the spatiotemporal features can be written as a sum of exponentials with origins (u_i, w_i) located at the spatial positions of the events.

$$F_e(u,w) = \sum_i \mathrm{e}^{\frac{u-u_i}{\tau}}\sigma(u_i - u)\delta(w - w_i) = \sum_i f_i(u)\delta_{w_i}(w) \tag{A.21}$$

The correlation from Eq. 4.11 then becomes

$$\begin{aligned}
\rho(e_L, e_R) &= \frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\sum_i f_i^L(u)\delta_{w_i}(w)\sum_i f_i^R(u)\delta_{w_i}(w)\mathrm{d}u\mathrm{d}w}{\sqrt{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\sum_i f_i^L(u)\delta_{w_i}(w)\right)^2\mathrm{d}u\mathrm{d}w \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\sum_i f_i^R(u)\delta_{w_i}(w)\right)^2\mathrm{d}u\mathrm{d}w}} \\
&= \frac{\sum_i\int_{-\infty}^{\infty} f_i^L(u) f_i^R(u)\mathrm{d}u}{\sqrt{\sum_i\int_{-\infty}^{\infty} f_i^L(u)^2\mathrm{d}u\sum_i\int_{-\infty}^{\infty} f_i^R(u)^2\mathrm{d}u}}
\end{aligned} \tag{A.22}$$

where

$$\begin{aligned} \int_{-\infty}^{\infty} f_i^L(u) f_i^R(u) \mathrm{d}u &= \int_{-\infty}^{0} \mathrm{e}^{u/\tau(1/v_T+1/v_R)} \mathrm{d}u \\ \int_{-\infty}^{\infty} f_i^L(u)^2 \mathrm{d}u &= \int_{-\infty}^{0} \mathrm{e}^{2u/v_L\tau} \\ \int_{-\infty}^{\infty} f_i^R(u)^2 \mathrm{d}u &= \int_{-\infty}^{0} \mathrm{e}^{2u/v_R\tau} \qquad , \quad \forall i \end{aligned} \tag{A.23}$$

Substituting the integrals from Eq. A.23 in Eq. A.22, the sums can be canceled and the same expression as stated in Eq. A.20 is obtained.

A.3.3 Effect of visual flow on event-based stereo matching

It is obvious that stereo matching based on the correlation of spatiotemporal features incorporates the motion of the scene. To what extent motion generally enhances the matching of exclusively spatial features using this approach is very difficult to show analytically. However, a simplified scenario can be analyzed, which provides further insight into this relationship. Consider the early approach to event-based processing, whereby the stream of events is accumulated over a short time interval to form static images, which are expressed in the form of the static time surface (STS) (see Eq. 4.7). When the STC algorithm is applied in this case, it only matches events based on their spatial features, regardless of their motion. Assume two events of the same polarity e_L and e_R, one from each camera, with equal features such that $\tilde{F}^L_{e_L} = \tilde{F}^R_{e_R}$. The tilde denotes that those are purely *spatial features*, meaning that they are obtained from the STS. In this case, the correlation defined in Eq. 4.11 equals 1 and is referred to as the *spatial correlation* $\tilde{\rho}(e_L, e_R)$. Generally, this means that the neighborhood does not provide any further information that could support or disprove the match. When compared to the correlation of the equivalent *spatiotemporal features* $F^L_{e_L}$ and $F^R_{e_R}$ of the ETS, the *spatiotemporal* correlation yields

$$\rho(e_L, e_R) = \rho(v_L, v_R) = 2\frac{\sqrt{v_L v_R}}{v_L + v_R} \tag{A.24}$$

This expression can be interpreted as a correlation between the magnitude of the projected velocities v_L and v_R of the features of an object moving in space at a velocity of $\mathbf{v}$ (see Appendix A.3.2 for derivation). It is interesting to see that the correlation of spatially equal features amounts to a simple correlation of their projected velocities. In a similar way, the spatiotemporal correlation of features from the LTS (defined as in Eq. 4.5) can be derived as

$$\rho(e_L, e_R) = \rho(v_L, v_R) = \frac{\min(v_L, v_R)}{\max(v_L, v_R)^2}\sqrt{v_L v_R} \tag{A.25}$$

The comparison between the two different approaches is illustrated in Fig. A.4b. The ETS is clearly less variant to differences in interocular velocity, which often occur in scenes where

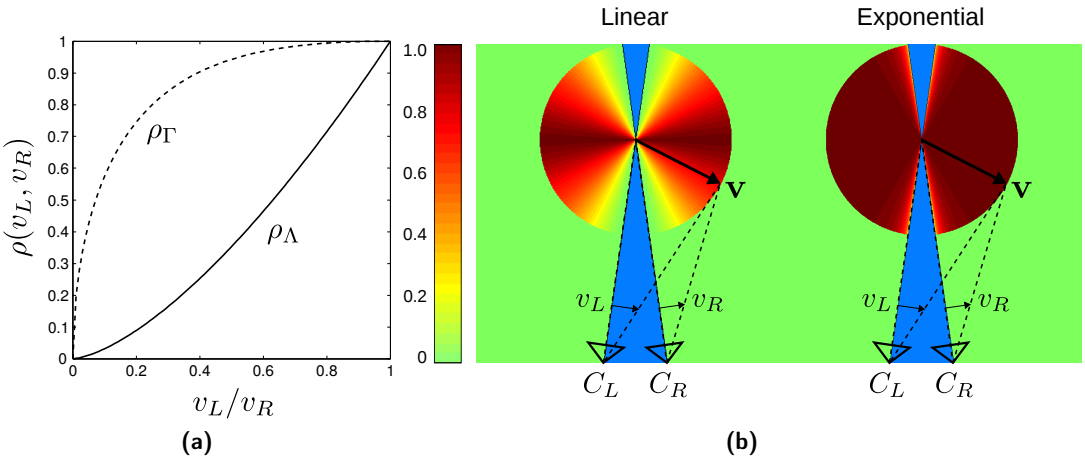


Figure A.4: Comparison of feature velocity correlation. **(a)** Correlation coefficient of velocities. **(b)** Comparison of velocity gradient in disparity space

objects move towards or away from the camera.

A.4 Modeling of the spike response

Given the assumptions $I_{syn} >> I_{\tau_{soma}}$ and $I_{mem} >> I_{th_{soma}}$, the dynamics of the neuron's circuit can be approximated with a linear first-order differential equation (Bartolozzi et al., 2006; Livi and Indiveri, 2009):

$$\tau_{soma}\frac{\mathrm{d}}{\mathrm{dt}}I_{mem}(t) + I_{mem}(t) = \frac{I_{th_{soma}}}{I_{\tau_{soma}}}I_{syn}(t) \tag{A.26}$$

with $\tau_{soma} = \frac{C_{mem}U_T}{kI_{\tau_{soma}}}$. Similarly, the equation of the synapse yields

$$\tau_{syn}\frac{\mathrm{d}}{\mathrm{dt}}I_{syn}(t) + I_{syn}(t) = \frac{I_{th_{syn}}}{I_{\tau_{syn}}}I_{in}(t) \tag{A.27}$$

with the same assumptions $I_{in} >> I_{\tau_{syn}}$ and $I_{syn} >> I_{th_{syn}}$ and $\tau_{syn} = \frac{C_{syn}U_T}{kI_{\tau_{syn}}}$. The bias currents I_{th} and I_τ can be modeled as

$$\begin{aligned} I_{th_{soma}} &= R_a \cdot a \\ I_{\tau_{soma}} &= R_b \cdot b \\ I_{th_{syn}} &= R_c \cdot c \\ I_{\tau_{syn}} &= R_d \cdot d \end{aligned} \tag{A.28}$$

where a, b, c and d are control parameters and R_a, R_b, R_c and R_d given constants. Rewriting equations A.26 and A.27 in state space yields

$$\begin{aligned} \dot{I}_{mem}(t) &= -\frac{R_b}{R_{soma}} \cdot b \cdot I_{mem}(t) + \frac{R_a}{R_{soma}} \cdot a \cdot I_{syn}(t) \\ \dot{I}_{syn}(t) &= -\frac{R_d}{R_{syn}} \cdot d \cdot I_{syn}(t) + \frac{R_c}{R_{syn}} \cdot c \cdot I_{in}(t) \end{aligned} \tag{A.29}$$

with $R_{soma} = \frac{C_{mem} U_T}{k}$ and $R_{syn} = \frac{C_{syn} U_T}{k}$. These differential equations can be easily solved by applying the Laplace transform

$$\begin{aligned} s \cdot I_{mem}(s) &= -\frac{R_b}{R_{soma}} \cdot b \cdot I_{mem}(s) + \frac{R_a}{R_{soma}} \cdot a \cdot I_{syn}(s) \\ s \cdot I_{syn}(s) &= -\frac{R_d}{R_{syn}} \cdot d \cdot I_{syn}(s) + \frac{R_c}{R_{syn}} \cdot c \cdot I_{in}(s) \end{aligned} \tag{A.30}$$

and can be solved for the following transfer functions:

$$\begin{aligned} G_{soma}(s) &= \frac{I_{mem}(s)}{I_{syn}(s)} = \frac{\frac{R_a}{R_{soma}} a}{s + \frac{R_b}{R_{soma}} b} = \frac{r_a a}{s + r_b b} \\ G_{syn}(s) &= \frac{I_{syn}(s)}{I_{in}(s)} = \frac{\frac{R_c}{R_{syn}} c}{s + \frac{R_d}{R_{syn}} d} = \frac{r_c c}{s + r_d d} \end{aligned} \tag{A.31}$$

These transfer functions correspond to simple linear, first-order low-pass filters. The overall transfer function can be found by multiplying each individual function together. This results in a linear, second-order low-pass filter.

$$\begin{aligned} G(s) &= \frac{I_{mem}(s)}{I_{in}(s)} = G_{soma}(s) \cdot G_{syn}(s) \\ &= \frac{r_a a}{s + r_b b} \cdot \frac{r_c c}{s + r_d d} \\ &= \frac{r_a r_c ac}{r_b r_d bd} \cdot \frac{r_b b r_d d}{(s + r_b b)(s + r_d d)} \\ &= K \frac{\alpha \beta}{(s + \alpha)(s + \beta)} \end{aligned} \tag{A.32}$$

In the last step, the substitutions $\alpha = r_b b$, $\beta = r_d d$ and $K = \frac{r_a r_c ac}{\alpha \beta}$ were made. It is noted that, although K is a function of α and β, this dependence can be neglected and K can be treated

as an individual parameter. This is because without loss of generality, a and c can be chosen such that they cancel the effect of α and β.

The step response $s(t)$ of a second-order low-pass filter in the form of Eq. A.32 can be expressed as follows:

$$s(t) = K\left(1 - \frac{\beta e^{-\alpha t} - \alpha e^{-\beta t}}{\beta - \alpha}\right) \tag{A.33}$$

As the system is linear, its response to a pulse can be obtained by summing the step response with a negative step response, shifted by the width ω of the pulse:

$$i_{mem}(t) = s(t) - s(t-\omega) = \begin{cases} K\left(1 - \frac{\beta e^{-\alpha t} - \alpha e^{-\beta t}}{\beta - \alpha}\right) & t \leq \omega \\ \frac{K}{\beta - \alpha}\left[\beta e^{-\alpha t}\left(e^{\alpha\omega} - 1\right) - \alpha e^{-\beta t}\left(e^{\beta\omega} - 1\right)\right] & t > \omega \end{cases} \tag{A.34}$$

A.4.1 Magnitude

When $t \leq \omega$, the response increases monotonically. As a result, the maximum is always reached in the region $t_{max} > \omega$. Deriving A.34 and setting it to zero yields:

$$t_{max} = \frac{1}{\alpha - \beta} \ln\left(\frac{e^{\alpha\omega} - 1}{e^{\beta\omega} - 1}\right) \tag{A.35}$$

The magnitude M of the response is equal to $i_{mem}(t_{max})$:

$$M = i_{mem}(t_{max}) = K\left(e^{\beta\omega} - 1\right)\left(\frac{e^{\alpha\omega} - 1}{e^{\beta\omega} - 1}\right)^{\frac{\beta}{\alpha-\beta}} \tag{A.36}$$

A.4.2 First-order approximation

If the poles α and β are well separated (usually $\frac{\beta}{\alpha} > 5$) and the magnitude is sufficiently high (small frequencies), Eq. A.32 can be approximated by a first-order low-pass filter with a pole located at $\min(\alpha, \beta)$.

$$G(s) \approx K\frac{\alpha}{s + \alpha} \tag{A.37}$$

The spike response can then be expressed as:

$$i_{mem}(t) = \begin{cases} K\left(1 - \mathrm{e}^{-\alpha t}\right) & t \leq \omega \\ K\mathrm{e}^{-\alpha t}\left(\mathrm{e}^{\alpha\omega} - 1\right) & t > \omega \end{cases} \tag{A.38}$$

A.4.3 Near-poles approximation

If the poles α and β are very close to each other, an approximation of the response is found by computing the limit $\lim_{\alpha\to\beta}(i_{mem}(t))$.

$$i_{mem}(t) = \begin{cases} K\left[1 - (\alpha t + 1)\,\mathrm{e}^{-\alpha t}\right] & t \leq \omega \\ K\left[\alpha t\left(\mathrm{e}^{\alpha\omega} - 1\right) - (\alpha\omega - 1)\mathrm{e}^{\alpha\omega} - 1\right]\mathrm{e}^{-\alpha t} & t > \omega \end{cases} \tag{A.39}$$

A.4.4 Duration

In the case of the first-order approximation, it can be observed that the response decays exponentially with a time constant that is equal to $\min(\alpha, \beta)$. In the case of the near-poles approximation, the tail is heavier but it also converges to an exponential decay with α. Therefore, a threshold e can be introduced, such that $\mathrm{e}^e \approx 0$, which defines the point after which the tail of the response is considered to have no effect any more. Thereafter, a measure of the duration of the response can be defined, which corresponds to the point in time t_e where $i_{mem}(t_e) = \mathrm{e}^e$. The duration T can then be expressed as follows:

$$T = t_e = \frac{e}{\alpha} + w \tag{A.40}$$

A.4.5 Normalization

The spike response can be normalized with respect to the width ω of the spike. Thus, the substitutions $\bar{\alpha} = \alpha\omega$, $\bar{\beta} = \beta\omega$ and $u = \frac{t}{\omega}$ are made.

$$i_{mem}(u) = \begin{cases} K\left(1 - \dfrac{\bar{\beta}\mathrm{e}^{-\bar{\alpha}u} - \bar{\alpha}\mathrm{e}^{-\bar{\beta}u}}{\bar{\beta} - \bar{\alpha}}\right) & u \leq 1 \\ \dfrac{K}{\bar{\beta} - \bar{\alpha}}\left[\bar{\beta}\mathrm{e}^{-\bar{\alpha}u}\left(\mathrm{e}^{\bar{\alpha}} - 1\right) - \bar{\alpha}\mathrm{e}^{-\bar{\beta}u}\left(\mathrm{e}^{\bar{\beta}} - 1\right)\right] & u > 1 \end{cases} \tag{A.41}$$

Bibliography

Adelstein, B. D., Lee, T. G., and Ellis, S. R. (2003). Head Tracking Latency in Virtual Environments: Psychophysics and a Model. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 47(20):2083–2087.

Alonso, J.-M. and Martinez, L. M. (1998). Functional connectivity between simple cells and complex cells in cat striate cortex. *Nature Neuroscience*, 1(5):395–403.

Ambrosch, K. and Kubinger, W. (2010). Accurate hardware-based stereo vision. *Computer Vision and Image Understanding*, 114(11):1303–1316.

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310.

Anzai, A., Ohzawa, I., and Freeman, R. D. (1999). Neural Mechanisms for Encoding Binocular Disparity: Receptive Field Position Versus Phase. *Journal of Neurophysiology*, 82(2):874–890.

Anzai, A., Ohzawa, I., and Freeman, R. D. (2001). Joint-encoding of motion and depth by visual cortical neurons: neural basis of the Pulfrich effect. *Nature Neuroscience*, 4(5):513–518.

Arnold, R. D. (1983). Automated Stereo Perception. Technical report.

Arthur, J. V. and Boahen, K. (2004). Recurrently connected silicon neurons with active dendrites for one-shot learning. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 3, pages 1699–1704. IEEE.

Backus, J. (1978). Can programming be liberated from the von Neumann style?: a functional style and its algebra of programs. *Communications of the ACM*, 21(8):613–641.

Bacon, B. A., Villemagne, J., Bergeron, A., Lepore, F., and Guillemot, J.-P. (1998). Spatial disparity coding in the superior colliculus of the cat. *Experimental Brain Research*, 119(3):333–344.

Baker, H. H. (1982). Depth from Edge and Intensity Based Stereo. Technical report.

Banks, M. S., Gepshtein, S., and Landy, M. S. (2004). Why Is Spatial Stereoresolution So Low? *The Journal of Neuroscience*, 24(9):2077–2089.

Banks, M. S., Gepshtein, S., and Rose, H. F. (2005). Local cross-correlation model of stereo correspondence. volume 5666, pages 53–61.

Barlow, H. B., Blakemore, C., and Pettigrew, J. D. (1967). The neural mechanism of binocular depth discrimination. *The Journal of Physiology*, 193(2):327–342.

Barnard, S. T. (1989). Stochastic stereo matching over scale. *International Journal of Computer Vision*, 3(1):17–32.

Bartolozzi, C. and Indiveri, G. (2007). Synaptic dynamics in analog VLSI. *Neural computation*, 19(10):2581–2603.

Bartolozzi, C., Mitra, S., and Indiveri, G. (2006). An ultra low power current-mode filter for neuromorphic systems and biomedical signal processing. In *IEEE Biomedical Circuits and Systems Conference, 2006. BioCAS 2006*, pages 130–133.

Becker, S. and Hinton, G. (1992). Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356):161–163.

Belbachir, A., Litzenberger, M., Schraml, S., Hofstatter, M., Bauer, D., Schon, P., Humenberger, M., Sulzbachner, C., Lunden, T., and Merne, M. (2012). CARE: A dynamic stereo vision sensor system for fall detection. In *2012 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 731–734.

Belbachir, A. N., Schraml, S., Mayerhofer, M., and Hofstatter, M. (2014). A Novel HDR Depth Camera for Real-Time 3d 360 #x00b0; Panoramic Vision. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 425–432.

Belhumeur, P. N. (1996). A Bayesian approach to binocular steropsis. *International Journal of Computer Vision*, 19(3):237–260.

Benjamin, B., Gao, P., McQuinn, E., Choudhary, S., Chandrasekaran, A., Bussat, J.-M., Alvarez-Icaza, R., Arthur, J., Merolla, P., and Boahen, K. (2014). Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations. *Proceedings of the IEEE*, 102(5):699–716.

Benosman, R., Clercq, C., Lagorce, X., Ieng, S.-H., and Bartolozzi, C. (2014). Event-Based Visual Flow. *IEEE Transactions on Neural Networks and Learning Systems*, 25(2):407–417.

Benosman, R., Ieng, S.-H., Rogister, P., and Posch, C. (2011). Asynchronous Event-Based Hebbian Epipolar Geometry. *Neural Networks, IEEE Transactions on*, 22(11):1723 –1734.

Bergen, J. R., Anandan, P., Hanna, K. J., and Hingorani, R. (1992). Hierarchical model-based motion estimation. In Sandini, G., editor, *Computer Vision — ECCV'92*, number 588 in Lecture Notes in Computer Science, pages 237–252. Springer Berlin Heidelberg.

Berner, R., Brandli, C., Yang, M., Liu, S.-C., and Delbruck, T. (2013). A 240x180 10mw 12us latency sparse-output vision sensor for mobile applications. In *2013 Symposium on VLSI Circuits (VLSIC)*, pages C186–C187.

Birchfield, S. and Tomasi, C. (1999). Depth Discontinuities by Pixel-to-Pixel Stereo. *International Journal of Computer Vision*, 35(3):269–293.

Black, M. J. and Rangarajan, A. (1996). On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision*, 19(1):57–91.

Blake, A. and Zisserman, A. (1987). *Visual Reconstruction*. The MIT Press, Cambridge, Massachussets.

Blake, R. and Wilson, H. (2011). Binocular vision. *Vision Research*, 51(7):754–770.

Bleyer, M. and Gelautz, M. (2005). A layered stereo matching algorithm using image segmentation and global visibility constraints. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(3):128–150.

Boahen, K. (2000). Point-to-point connectivity between neuromorphic chips using address events. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 47(5):416–434.

Boahen, K. A. (1997). *Retinomorphic vision systems: Reverse engineering the vertebrate retina*. PhD thesis, California Institute of Technology.

Bobick, A. F. and Intille, S. S. (1999). Large Occlusion Stereo. *International Journal of Computer Vision*, 33(3):181–200.

Boegerhausen, M., Suter, P., and Liu, S.-C. (2003). Modeling Short-Term Synaptic Depression in Silicon. *Neural Computation*, 15(2):331–348.

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239.

Brader, J. M., Senn, W., and Fusi, S. (2007). Learning real-world stimuli in a neural network with spike-driven synaptic dynamics. *Neural computation*, 19(11):2881–2912.

Bradshaw, M. F. and Cumming, B. G. (1997). The direction of retinal motion facilitates binocular steropsis. *Proceedings of the Royal Society of London B: Biological Sciences*, 264(1387):1421–1427.

Brandli, C., Berner, R., Yang, M., Liu, S.-C., and Delbruck, T. (2014a). A 240x180 130 dB 3us Latency Global Shutter Spatiotemporal Vision Sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341.

Brandli, C., Mantel, T. A., Hutter, M., and Delbruck, T. (2014b). Adaptive Pulsed Laser Line Extraction for Terrain Reconstruction using a Dynamic Vision Sensor. *Frontiers in Neuromorphic Engineering*, 7:275.

Bredfeldt, C. E. and Cumming, B. G. (2006). A Simple Account of Cyclopean Edge Responses in Macaque V2. *The Journal of Neuroscience*, 26(29):7581–7596.

Brette, R. and Gerstner, W. (2005). Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity. *Journal of Neurophysiology*, 94(5):3637–3642.

Brooks, K. R. (2002). Interocular velocity difference contributes to stereomotion speed perception. *Journal of Vision*, 2(3):2–2.

Brändli, C. P. (2015). *Event-Based Machine Vision*. PhD thesis, Diss., Eidgenössische Technische Hochschule ETH Zürich, Nr. 22500.

Burt, P. and Julesz, B. (1980). A disparity gradient limit for binocular fusion. *Science*, 208(4444):615–617.

Camunas-Mesa, L., Acosta-Jiménez, A., Serrano-Gotarredona, T., and Linares-Barranco, A. (2008). Fully digital AER convolution chip for vision processing. In *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, pages 652–655. IEEE.

Camunas-Mesa, L. A., Serrano-Gotarredona, T., Ieng, S. H., Benosman, R. B., and Linares-Barranco, B. (2014). On the use of orientation filters for 3d reconstruction in event-driven stereo vision. *Frontiers in Neuroscience*, 8.

Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698.

Cardinale, J. (2006). *Tracking objects and wing beat analysis methods of a fruit fly with the event-based silicon retina*. Semester Thesis, ETH, Zurich, Switzerland.

Carneiro, J., Ieng, S.-H., Posch, C., and Benosman, R. (2013). Event-based 3d reconstruction from neuromorphic retinas. *Neural Networks*, 45:27–38.

Censi, A. and Scaramuzza, D. (2014). Low-latency event-based visual odometry. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 703–710.

Censi, A., Strubel, J., Brandli, C., Delbruck, T., and Scaramuzza, D. (2013). Low-latency localization by active LED markers tracking using a dynamic vision sensor. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 891–898.

Choi, T. Y. W., Shi, B. E., Boahen, K., and others (2004). An on-off orientation selective address event representation image transceiver chip. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 51(2):342–353.

Cogan, A. I., Lomakin, A. J., and Rossi, A. F. (1993). Depth in anticorrelated stereograms: effects of spatial density and interocular delay. *Vision Research*, 33(14):1959–1975.

Cormack, L. K., Stevenson, S. B., and Schor, C. M. (1993). Disparity-tuned channels of the human visual system. *Visual Neuroscience*, 10(04):585–596.

Cottereau, B. R., McKee, S. P., and Norcia, A. M. (2014). Dynamics and cortical distribution of neural responses to 2d and 3d motion in human. *Journal of Neurophysiology*, 111(3):533–543.

Cox, I. J., Hingorani, S. L., Rao, S. B., and Maggs, B. M. (1996). A Maximum Likelihood Stereo Algorithm. *Computer Vision and Image Understanding*, 63(3):542–567.

Culurciello, E., Etienne-Cummings, R., Boahen, K., and others (2003). A biomorphic digital image sensor. *Solid-State Circuits, IEEE Journal of*, 38(2):281–294.

Cumming, B. G. and DeAngelis, G. C. (2001). The Physiology of Stereopsis. *Annual Review of Neuroscience*, 24(1):203–238.

Cumming, B. G. and Parker, A. J. (1994). Binocular mechanisms for detecting motion-in-depth. *Vision Research*, 34(4):483–495.

Cumming, B. G. and Parker, A. J. (1997). Responses of primary visual cortical neurons to binocular disparity without depth perception. *Nature*, 389(6648):280–283.

Cumming, B. G. and Parker, A. J. (1999). Binocular Neurons in V1 of Awake Monkeys Are Selective for Absolute, Not Relative, Disparity. *The Journal of Neuroscience*, 19(13):5602–5618.

Cumming, B. G. and Parker, A. J. (2000). Local Disparity Not Perceived Depth Is Signaled by Binocular Neurons in Cortical Area V1 of the Macaque. *The Journal of Neuroscience*, 20(12):4758–4767.

DeAngelis, G. C. (2000). Seeing in three dimensions: the neurophysiology of stereopsis. *Trends in Cognitive Sciences*, 4(3):80–90.

DeAngelis, G. C., Ohzawa, I., and Freeman, R. D. (1991). Depth is encoded in the visual cortex by a specialized receptive field structure. *Nature*, 352(6331):156–159.

Deiss, S. R., Douglas, R. J., Whatley, A. M., and others (1999). A pulse-coded communications infrastructure for neuromorphic systems. *Pulsed neural networks*, pages 157–178.

Delbruck, T., Berner, R., Lichtsteiner, P., and Dualibe, C. (2010a). 32-bit Configurable bias current generator with sub-off-current capability. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1647–1650.

Delbruck, T. and Lichtsteiner, P. (2007). Fast sensory motor control based on event-based hybrid neuromorphic-procedural system. In *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pages 845–848. IEEE.

Delbruck, T., Linares-Barranco, B., Culurciello, E., and Posch, C. (2010b). Activity-driven, event-based vision sensors. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2426–2429.

Delbruck, T. and Mead, C. (1994). Adaptive photoreceptor with wide dynamic range. In *, 1994 IEEE International Symposium on Circuits and Systems, 1994. ISCAS '94*, volume 4, pages 339–342 vol.4.

Delbruck, T. and Oberhoff, D. (2004). Self-biasing low power adaptive photoreceptor. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*, volume 4, pages IV–844. IEEE.

Destexhe, A., Mainen, Z. F., and Sejnowski, T. J. (1998). Kinetic models of synaptic transmission. *Methods in neuronal modeling*, 2:1–25.

Dev, P. (1974). Segmentation processes in visual perception: A cooperative neural model. Technical Report, University of Massachusetts at Amherst.

Dodd, J. V., Krug, K., Cumming, B. G., and Parker, A. J. (2001). Perceptually Bistable Three-Dimensional Figures Evoke High Choice Probabilities in Cortical Area MT. *The Journal of Neuroscience*, 21(13):4809–4821.

Dominguez-Morales, M. J., Cerezuela-Escudero, E., Perez-Peña, F., Jimenez-Fernandez, A., Linares-Barranco, A., and Jimenez-Moreno, G. (2013). On the AER Stereo-Vision Processing: A Spike Approach to Epipolar Matching. In Lee, M., Hirose, A., Hou, Z.-G., and Kil, R. M., editors, *Neural Information Processing*, number 8226 in Lecture Notes in Computer Science, pages 267–275. Springer Berlin Heidelberg.

Drazen, D., Lichtsteiner, P., Häfliger, P., Delbrück, T., and Jensen, A. (2011). Toward real-time particle tracking using an event-based dynamic vision sensor. *Experiments in Fluids*, 51(5):1465–1469.

Drazic, V. and Sabater, N. (2012). A Precise Real-time Stereo Algorithm. In *Proceedings of the 27th Conference on Image and Vision Computing New Zealand*, IVCNZ '12, pages 138–143, New York, NY, USA. ACM.

Drumheller, M. and Poggio, T. (1986). On parallel stereo. In *1986 IEEE International Conference on Robotics and Automation. Proceedings*, volume 3, pages 1439–1448.

Dupeyron, D., Le Masson, S., Deval, Y., Le Masson, G., and Dom, J.-P. (1996). A BiCMOS implementation of the Hodgkin-Huxley formalism. In *Microelectronics for Neural Networks, 1996., Proceedings of Fifth International Conference on*, pages 311–316. IEEE.

Edwards, M. and Schor, C. M. (1999). Depth aliasing by the transient-stereopsis system. *Vision Research*, 39(26):4333–4340.

Edwards, R. T. and Cauwenberghs, G. (2000). Synthesis of log-domain filters from first-order building blocks. In *Research perspectives on dynamic translinear and log-domain circuits*, pages 71–80. Springer.

Egnal, G. and Wildes, R. (2002). Detecting binocular half-occlusions: empirical comparisons of five approaches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1127–1133.

Eibensteiner, F., Kogler, J., and Scharinger, J. (2014). A High-Performance Hardware Architecture for a Frameless Stereo Vision Algorithm Implemented on a FPGA Platform. pages 623–630.

Eibensteiner, F., Kogler, J., Sulzbachner, C., and Scharinger, J. (2012). Stereo-Vision Algorithm Based on Bio-Inspired Silicon Retinas for Implementation in Hardware. In Moreno-Díaz, R., Pichler, F., and Quesada-Arencibia, A., editors, *Computer Aided Systems Theory – EUROCAST 2011*, number 6927 in Lecture Notes in Computer Science, pages 624–631. Springer Berlin Heidelberg.

Farabet, C., Martini, B., Corda, B., Akselrod, P., Culurciello, E., and LeCun, Y. (2011). NeuFlow: A runtime reconfigurable dataflow processor for vision. In *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 109–116.

Farquhar, E. and Hasler, P. (2005). A bio-physically inspired silicon neuron. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 52(3):477–488.

Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., and others (2010). Building Watson: An overview of the DeepQA project. *AI magazine*, 31(3):59–79.

Filippini, H. R. and Banks, M. S. (2009). Limits of stereopsis explained by local cross-correlation. *Journal of Vision*, 9(1):8.

Firouzi, M. and Conradt, J. (2015). Asynchronous Event-based Cooperative Stereo Matching Using Neuromorphic Silicon Retinas. *Neural Processing Letters*, pages 1–16.

Fleet, D. J., Wagner, H., and Heeger, D. J. (1996). Neural encoding of binocular disparity: Energy models, position shifts and phase shifts. *Vision Research*, 36(12):1839–1857.

Folowosele, F., Etienne-Cummings, R., and Hamilton, T. J. (2009). A CMOS switched capacitor implementation of the Mihalas-Niebur neuron. In *Biomedical Circuits and Systems Conference, 2009. BioCAS 2009. IEEE*, pages 105–108. IEEE.

Fortune, E. S. and Rose, G. J. (2001). Short-term synaptic plasticity as a temporal filter. *Trends in Neurosciences*, 24(7):381–385.

Freeman, R. D. and Ohzawa, I. (1990). On the neurophysiological organization of binocular vision. *Vision Research*, 30(11):1661–1676.

Freeman, W. T., Pasztor, E. C., and Carmichael, O. T. (2000). Learning Low-Level Vision. *International Journal of Computer Vision*, 40(1):25–47.

Frey, D. (2000). Future implications of the log domain paradigm. In *Circuits, Devices and Systems, IEE Proceedings-*, volume 147, pages 65–72. IET.

Fua, P. (1993). A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49.

Furber, S., Galluppi, F., Temple, S., and Plana, L. (2014). The SpiNNaker Project. *Proceedings of the IEEE*, 102(5):652–665.

Fusi, S., Annunziato, M., Badoni, D., Salamon, A., and Amit, D. J. (2000). Spike-driven synaptic plasticity: theory, simulation, VLSI implementation. *Neural computation*, 12(10):2227–2258.

Gallup, D., Frahm, J.-M., Mordohai, P., and Pollefeys, M. (2008). Variable baseline/resolution stereo. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, pages 1–8.

Gamble, E. and Poggio, T. (1987). Visual Integration and Detection of Discontinuities: The Key Role of Intensity Edges. Technical report.

Geiger, D. and Girosi, F. (1989). Parallel and Deterministic Algorithms for MRFs: Surface Reconstruction and Integration.

Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741.

Georgoulas, C., Kotoulas, L., Sirakoulis, G. C., Andreadis, I., and Gasteratos, A. (2008). Real-time disparity map computation module. *Microprocessors and Microsystems*, 32(3):159–170.

Gerstner, W. and Kistler, W. M. (2002). *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press.

Gollisch, T. and Meister, M. (2010). Eye smarter than scientists believed: neural computations in circuits of the retina. *Neuron*, 65(2):150–164.

Graybiel, A. M. (1976). Evidence for banding of the cat's ipsilateral retinotectal connection. *Brain Research*, 114(2):318–327.

Haefner, R. M. and Cumming, B. G. (2008). Adaptation to Natural Binocular Disparities in Primate V1 Explained by a Generalized Energy Model. *Neuron*, 57(1):147–158.

Hammond, P. (1991). Binocular phase specificity of striate cortical neurones. *Experimental Brain Research*, 87(3):615–623.

Hannah, M. J. (1974). Computer Matching of Areas in Stereo Images. Technical report.

Hariyama, M., Sasaki, H., and Kameyama, M. (2004). Architecture of a stereo matching VLSI processor based on hierarchically parallel memory access. In *The 2004 47th Midwest Symposium on Circuits and Systems, 2004. MWSCAS '04*, volume 2, pages II–245–II–247 vol.2.

Hartley, R. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition.

Hasler, J. and Marr, B. (2013). Finding a roadmap to achieve large neuromorphic hardware systems. *Frontiers in neuroscience*, 7.

Hasler, P. and Lande, T. S. (2001). Overview of floating-gate devices, circuits, and systems. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 48(1):1–3.

Hess, P. (2006). Low-level Stereo Matching using Event-based Silicon Retina, Semester Project, ETH Zurich.

Hibbard, P. B. (2008). Binocular energy responses to natural images. *Vision Research*, 48(12):1427–1439.

Hinkle, D. A. and Connor, C. E. (2002). Three-dimensional orientation tuning in macaque area V4. *Nature Neuroscience*, 5(7):665–670.

Hinkle, D. A. and Connor, C. E. (2005). Quantitative Characterization of Disparity Tuning in Ventral Pathway Area V4. *Journal of Neurophysiology*, 94(4):2726–2737.

Hirschmuller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*, volume 2, pages 807–814 vol. 2.

Hirschmuller, H. (2008). Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341.

Hirschmuller, H. and Scharstein, D. (2007). Evaluation of Cost Functions for Stereo Matching. In *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, pages 1–8.

Hong, L. and Chen, G. (2004). Segment-based stereo matching using graph cuts. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*, volume 1, pages I–74–I–81 Vol.1.

Howard, I. P. (2002). *Seeing in depth, Vol. 1: Basic mechanisms*, volume v. University of Toronto Press, Toronto, ON, Canada.

Howard, I. P. and Rogers, B. J. (2012). *Perceiving in Depth, Volume 2: Stereoscopic Vision*. OUP USA.

Hsu, F.-H. (2002). *Behind Deep Blue: Building the computer that defeated the world chess champion*. Princeton University Press.

Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154.2.

Hubel, D. H. and Wiesel, T. N. (1970). Stereoscopic Vision in Macaque Monkey: Cells sensitive to Binocular Depth in Area 18 of the Macaque Monkey Cortex. *Nature*, 225(5227):41–42.

Hynna, K. M. and Boahen, K. (2007). Thermodynamically equivalent silicon models of voltage-dependent ion channels. *Neural Computation*, 19(2):327–350.

Indiveri, G. (2002). Neuromorphic bistable VLSI synapses with spike-timing-dependent plasticity. In *NIPS*, pages 1091–1098.

Indiveri, G. (2007). Synaptic plasticity and spike-based computation in VLSI networks of integrate-and-fire neurons. *Neural Inf. Process. Lett. Rev*, 11:135–146.

Indiveri, G., Chicca, E., and Douglas, R. (2006). A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity. *Neural Networks, IEEE Transactions on*, 17(1):211–221.

Indiveri, G., Linares-Barranco, B., Hamilton, T. J., van Schaik, A., Etienne-Cummings, R., Delbruck, T., Liu, S.-C., Dudek, P., Häfliger, P., Renaud, S., Schemmel, J., Cauwenberghs, G., Arthur, J., Hynna, K., Folowosele, F., Saighi, S., Serrano-Gotarredona, T., Wijekoon, J., Wang, Y., and Boahen, K. (2011). Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience*, 5.

Indiveri, G., Stefanini, F., and Chicca, E. (2010). Spike-based learning with a generalized integrate and fire silicon neuron. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1951–1954.

Izhikevich, E. M. and others (2003). Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572.

Janssen, P., Vogels, R., Liu, Y., and Orban, G. A. (2003). At Least at the Level of Inferior Temporal Cortex, the Stereo Correspondence Problem Is Solved. *Neuron*, 37(4):693–701.

Jenkin, M. R. M., Jepson, A. D., John, and Tsotsos, K. (1991). Techniques for disparity measurement. *CVGIP: IU*.

Jia, Y., Xu, Y., Liu, W., Yang, C., Zhu, Y., Zhang, X., and An, L. (2003). A Miniature Stereo Vision Machine for Real-Time Dense Depth Mapping. In Crowley, J. L., Piater, J. H., Vincze, M., and Paletta, L., editors, *Computer Vision Systems*, number 2626 in Lecture Notes in Computer Science, pages 268–277. Springer Berlin Heidelberg. DOI: 10.1007/3-540-36592-3_26.

Jolivet, R., Lewis, T. J., and Gerstner, W. (2004). Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy. *Journal of neurophysiology*, 92(2):959–976.

Jones, D. G. and Malik, J. (1992). A computational framework for determining stereo correspondence from a set of linear spatial filters. In Sandini, G., editor, *Computer Vision — ECCV'92*, number 588 in Lecture Notes in Computer Science, pages 395–410. Springer Berlin Heidelberg.

Jones, J. P. and Palmer, L. A. (1987). The two-dimensional spatial structure of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1187–1211.

Joshi, S., Deiss, S., Arnold, M., Park, J., Yu, T., and Cauwenberghs, G. (2010). Scalable event routing in hierarchical neural array architecture with global synaptic connectivity. In *Cellular Nanoscale Networks and Their Applications (CNNA), 2010 12th International Workshop on*, pages 1–6. IEEE.

Julesz, B. (1960). Binocular depth perception of computer-generated patterns. *Bell System Technical Journal*, 39:1125–1162.

Kalomiros, J. and Lygouras, J. (2010). Comparative Study of Local SAD and Dynamic Programming for Stereo Processing Using Dedicated Hardware. *EURASIP Journal on Advances in Signal Processing*, 2009(1):914186.

Kameda, S. and Yagi, T. (2003). An analog VLSI chip emulating sustained and transient response channels of the vertebrate retina. *Neural Networks, IEEE Transactions on*, 14(5):1405–1412.

Kanade, T. and Okutomi, M. (1994). A stereo matching algorithm with an adaptive window: theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932.

Kanade, T., Yoshida, A., Oda, K., Kano, H., and Tanaka, M. (1996). A stereo machine for video-rate dense depth mapping and its new applications. In *Proceedings CVPR '96, 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996*, pages 196–202.

Karmazin, R., Otero, C. T. O., and Manohar, R. (2013). celltk: Automated layout for asynchronous circuits with nonstandard cells. In *Asynchronous Circuits and Systems (ASYNC), 2013 IEEE 19th International Symposium on*, pages 58–66. IEEE.

Kass, M. (1988). Linear image features in stereopsis. *International Journal of Computer Vision*, 1(4):357–368.

Khan, M., Lester, D., Plana, L., Rast, A., Jin, X., Painkras, E., and Furber, S. (2008). SpiNNaker: Mapping neural networks onto a massively-parallel chip multiprocessor. In *IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence)*, pages 2849–2856.

Kogler, J., Eibensteiner, F., Humenberger, M., Gelautz, M., and Scharinger, J. (2013). Ground Truth Evaluation for Event-Based Silicon Retina Stereo Data. In *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 649–656.

Kogler, J., Eibensteiner, F., Humenberger, M., Sulzbachner, C., Gelautz, M., and Scharinger, J. (2014). Enhancement of sparse silicon retina-based stereo matching using belief propagation and two-stage postfiltering. *Journal of Electronic Imaging*, 23(4):043011–043011.

Kogler, J., Humenberger, M., and Sulzbachner, C. (2011). Event-Based Stereo Matching Approaches for Frameless Address Event Stereo Data. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Wang, S., Kyungnam, K., Benes, B., Moreland, K., Borst, C., DiVerdi, S., Yi-Jen, C., and

Ming, J., editors, *Advances in Visual Computing*, number 6938 in Lecture Notes in Computer Science, pages 674–685. Springer Berlin Heidelberg.

Kogler, J., Sulzbachner, C., and Kubinger, W. (2009). Bio-inspired Stereo Vision System with Silicon Retina Imagers. In Fritz, M., Schiele, B., and Piater, J. H., editors, *Computer Vision Systems*, number 5815 in Lecture Notes in Computer Science, pages 174–183. Springer Berlin Heidelberg.

Kolmogorov, V. and Zabih, R. (2001). Computing visual correspondence with occlusions using graph cuts. In *Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings*, volume 2, pages 508–515 vol.2.

Kowalczuk, J., Psota, E., and Perez, L. (2013). Real-Time Stereo Matching on CUDA Using an Iterative Refinement Method for Adaptive Support-Weight Correspondences. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(1):94–104.

Kramer, J. (2002). An on/off transient imager with event-driven, asynchronous read-out. In *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, volume 2, pages II–165. IEEE.

Krug, K., Cumming, B. G., and Parker, A. J. (2004). Comparing Perceptual Signals of Single V5/MT Neurons in Two Binocular Depth Tasks. *Journal of Neurophysiology*, 92(3):1586–1596.

Kuhn, M., Moser, S., Isler, O., Gurkaynak, F., Burg, A., Felber, N., Kaeslin, H., and Fichtner, W. (2003). Efficient ASIC implementation of a real-time depth mapping stereo vision system. In *2003 IEEE 46th Midwest Symposium on Circuits and Systems*, volume 3, pages 1478–1481 Vol. 3.

Lazaros, N., Sirakoulis, G. C., and Gasteratos, A. (2008). Review of Stereo Vision Algorithms: From Software to Hardware. *International Journal of Optomechatronics*, 2(4):435–462.

Lazzaro, J., Wawrzynek, J., and Kramer, A. (1994). Systems technologies for silicon auditory models. *Micro, IEEE*, 14(3):7–15.

Lazzaro, J., Wawrzynek, J., Mahowald, M., Sivilotti, M., and Gillespie, D. (1993). Silicon auditory processors as computer peripherals. *IEEE Transactions on Neural Networks*, 4(3):523–528.

Lee, J., Delbruck, T., Park, P. K., Pfeiffer, M., Shin, C.-W., Ryu, H., and Kang, B. C. (2012). Live demonstration: Gesture-Based remote control using stereo pair of dynamic vision sensors. In *2012 IEEE International Symposium on Circuits and Systems*.

Lee, J. C. (2007). Head Tracking for Desktop VR Displays using the Wii Remote.

Lenero-Bardallo, J., Serrano-Gotarredona, T., and Linares-Barranco, B. (2010). A signed spatial contrast event spike retina chip. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2438 –2441.

Lenero-Bardallo, J. A., Serrano-Gotarredona, T., and Linares-Barranco, B. (2011). A 3.6 us latency asynchronous frame-free event-driven dynamic-vision-sensor. *Solid-State Circuits, IEEE Journal of*, 46(6):1443–1455.

Lichtsteiner, P., Delbruck, T., and Kramer, J. (2004). Improved ON/OFF temporally differentiating address-event imager. In *Electronics, Circuits and Systems, 2004. ICECS 2004. Proceedings of the 2004 11th IEEE International Conference on*, pages 211–214. IEEE.

Lichtsteiner, P., Posch, C., and Delbruck, T. (2008). A 128x128 120 dB 15 us Latency Asynchronous Temporal Contrast Vision Sensor. *Solid-State Circuits, IEEE Journal of*, 43(2):566 –576.

Likova, L. T. and Tyler, C. W. (2007). Stereomotion processing in the human occipital cortex. *NeuroImage*, 38(2):293–305.

Lippert, J., Fleet, D. J., and Wagner, H. (2000). Disparity tuning as simulated by a neural net. *Biological Cybernetics*, 83(1):61–72.

Litzenberger, M., Belbachir, A., Schon, P., and Posch, C. (2007). Embedded Smart Camera for High Speed Vision. In *First ACM/IEEE International Conference on Distributed Smart Cameras, 2007. ICDSC '07*, pages 81–86.

Liu, S.-C., Delbrück, T., Indiveri, G., Whatley, A. M., and Douglas, R. J. (2015). *Event-Based Neuromorphic Systems*. John Wiley & Sons.

Liu, S.-C., Kramer, J., Indiveri, G., Delbrück, T., and Douglas, R. J. (2002). *Analog VLSI: circuits and principles*. MIT press.

Liu, Y., Vogels, R., and Orban, G. A. (2004). Convergence of Depth from Texture and Depth from Disparity in Macaque Inferior Temporal Cortex. *The Journal of Neuroscience*, 24(15):3795–3800.

Livi, P. and Indiveri, G. (2009). A current-mode conductance-based silicon neuron for address-event neuromorphic systems. In *IEEE International Symposium on Circuits and Systems, 2009. ISCAS 2009*, pages 2898–2901.

Livingstone, M. S. and Tsao, D. Y. (1999). Receptive fields of disparity-selective neurons in macaque striate cortex. *Nature Neuroscience*, 2(9):825–832.

London, M. and Häusser, M. (2005). Dendritic Computation. *Annual Review of Neuroscience*, 28(1):503–532.

Maass, W. and Sontag, E. D. (2000). Neural systems as nonlinear filters. *Neural Computation*, 12(8):1743–1772.

Mahowald, M. (1994a). Analog VLSI chip for stereocorrespondence. In *Circuits and Systems, 1994. ISCAS '94., 1994 IEEE International Symposium on*, volume 6, pages 347 –350 vol.6.

Mahowald, M. (1994b). *An analog VLSI system for stereoscopic vision*. Springer Science & Business Media.

Mahowald, M. and Douglas, R. (1991). A silicon neuron.

Mahowald, M. A. and Delbrück, T. (1989). Cooperative Stereo Matching Using Static and Dynamic Image Features. In Mead, C. and Ismail, M., editors, *Analog VLSI Implementation of Neural Systems*, number 80 in The Kluwer International Series in Engineering and Computer Science, pages 213–238. Springer US.

Mania, K., Adelstein, B. D., Ellis, S. R., and Hill, M. I. (2004). Perceptual Sensitivity to Head Tracking Latency in Virtual Environments with Varying Degrees of Scene Complexity. In *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization*, APGV '04, pages 39–47, New York, NY, USA. ACM.

Manohar, R. (2006). Reconfigurable asynchronous logic. In *Proceedings of IEEE Custom Integrated Circuits Conference*, pages 13–20.

Marr, D. (1982). *Vision*. The MIT Press, Cambridge, Massachussets.

Marr, D. and Poggio, T. (1976). Cooperative computation of stereo disparity. *Science*, 194(4262):283–287.

Marr, D. and Poggio, T. (1979). A Computational Theory of Human Stereo Vision. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 204(1156):301–328.

Marroquin, J. L. (1983). Design of Cooperative Networks. Working Paper, MIT Artificial Intelligence Laboratory.

Martin, A. J., Lines, A., Manohar, R., Nystroem, M., Penzes, P., Southworth, R., and Cummings, U. (1997). The design of an asynchronous MIPS R3000 microprocessor. In *arvlsi*, page 164. IEEE.

Matolin, D., Posch, C., and Wohlgenannt, R. (2009). True correlated double sampling and comparator design for time-based image sensors. In *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pages 1269–1272. IEEE.

Mayoral, R., Lera, G., and Pérez-Ilzarbe, M. J. (2006). Evaluation of correspondence errors for stereo. *Image and Vision Computing*, 24(12):1288–1300.

McDonnell, M. J. (1981). Box-filtering techniques. *Computer Graphics and Image Processing*, 17(1):65–70.

Mead, C. (1989). *Analog VLSI and neural systems*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Merolla, P., Arthur, J., Akopyan, F., Imam, N., Manohar, R., and Modha, D. S. (2011). A digital neurosynaptic core using embedded crossbar memory with 45pj per spike in 45nm. In *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pages 1–4. IEEE.

Merolla, P. and Boahen, K. A. (2003). A recurrent model of orientation maps with simple and complex cells. *Departmental Papers (BE)*, page 26.

Merolla, P. A., Arthur, J. V., Alvarez-Icaza, R., Cassidy, A. S., Sawada, J., Akopyan, F., Jackson, B. L., Imam, N., Guo, C., Nakamura, Y., Brezzo, B., Vo, I., Esser, S. K., Appuswamy, R., Taba, B., Amir, A., Flickner, M. D., Risk, W. P., Manohar, R., and Modha, D. S. (2014). A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Mihalas, S. and Niebur, E. (2009). A generalized linear integrate-and-fire neural model produces diverse spiking behaviors. *Neural computation*, 21(3):704–718.

Mimeault, D., Paquet, V., Molotchnikoff, S., Lepore, F., and Guillemot, J.-P. (2004). Disparity sensitivity in the superior colliculus of the cat. *Brain Research*, 1010(1–2):87–94.

Moini, A. (2000). *Vision Chips*. Springer.

Mueggler, E., Baumli, N., Fontana, F., and Scaramuzza, D. (2015). Towards evasive maneuvers with quadrotors using dynamic vision sensors. In *Mobile Robots (ECMR), 2015 European Conference on*, pages 1–8. IEEE.

Mueggler, E., Huber, B., and Scaramuzza, D. (2014). Event-based, 6-DOF pose tracking for high-speed maneuvers. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pages 2761–2768.

Muller, G. and Conradt, J. (2011). A miniature low-power sensor system for real time 2d visual tracking of LED markers. In *2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2429–2434.

Mühlmann, K., Maier, D., Hesser, J., and Männer, R. (2002). Calculating Dense Disparity Maps from Color Stereo Images, an Efficient Implementation. *International Journal of Computer Vision*, 47(1-3):79–88.

Neftci, E. and Indiveri, G. (2010). A device mismatch compensation method for VLSI neural networks. In *2010 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 262–265.

Ni, Z., Bolopion, A., Agnus, J., Benosman, R., and Regnier, S. (2012). Asynchronous Event-Based Visual Shape Tracking for Stable Haptic Feedback in Microrobotics. *IEEE Transactions on Robotics*, 28(5):1081–1089.

Nienborg, H., Bridge, H., Parker, A. J., and Cumming, B. G. (2004). Receptive field size in V1 neurons limits acuity for perceiving disparity modulation. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 24(9):2065–2076.

Nishihara, H. K. (1984). Practical Real-Time Imaging Stereo Matcher. *Optical Engineering*, 23(5):235536–235536–.

Normann, R. A. and Perlman, I. (1979). The effects of background illumination on the photoresponses of red and green cones. *The Journal of Physiology*, 286(1):491–507.

Ohta, Y. and Kanade, T. (1985). Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(2):139–154.

Ohzawa, I. (1998). Mechanisms of stereoscopic vision: the disparity energy model. *Current Opinion in Neurobiology*, 8(4):509–515.

Ohzawa, I., DeAngelis, G. C., and Freeman, R. D. (1990). Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors. *Science*, 249(4972):1037–1041.

Ohzawa, I., Deangelis, G. C., and Freeman, R. D. (1997). Encoding of Binocular Disparity by Complex Cells in the Cat's Visual Cortex. *Journal of Neurophysiology*, 77(6):2879–2909.

Okutomi, M. and Kanade, T. (1992). A locally adaptive window for signal matching. *International Journal of Computer Vision*, 7(2):143–162.

Osswald, M. (2011). Biologically Inspired Stereo Vision on Neuromorphic Hardware, Research Proposal, University of Zurich.

Pack, C. C., Born, R. T., and Livingstone, M. S. (2003). Two-Dimensional Substructure of Stereo and Motion Interactions in Macaque Visual Cortex. *Neuron*, 37(3):525–535.

Park, S. and Jeong, H. (2007). Real-time Stereo Vision FPGA Chip with Low Error Rate. In *International Conference on Multimedia and Ubiquitous Engineering, 2007. MUE '07*, pages 751–756.

Peng, S., Fang, D., Teifel, J., and Manohar, R. (2005). Automated synthesis for asynchronous FPGAs. In *Proceedings of the 2005 ACM/SIGDA 13th international symposium on Field-programmable gate arrays*, pages 163–173. ACM.

Philipp, R. M. (2009). *Single-chip integration of a 3-D imaging system*. PhD thesis, The Johns Hopkins University.

Piatkowska, E., Belbachir, A., and Gelautz, M. (2013). Asynchronous Stereo Vision for Event-Driven Dynamic Stereo Sensor Using an Adaptive Cooperative Approach. In *2013 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 45–50.

Piatkowska, E., Belbachir, A. N., and Gelautz, M. (2014). Cooperative and asynchronous stereo vision for dynamic vision sensors. *Measurement Science and Technology*, 25(5):055108.

Poggio, G. F. (1991). Physiological basis of stereoscopic vision. *Vision and visual dysfunction. Binocular vision*, pages 224–238.

Poggio, G. F., Motter, B. C., Squatrito, S., and Trotter, Y. (1985a). Responses of neurons in visual cortex (V1 and V2) of the alert macaque to dynamic random-dot stereograms. *Vision Research*, 25(3):397–406.

Poggio, T., Torre, V., and Koch, C. (1985b). Computational vision and regularization theory. *Nature*, 317(6035):314–319.

Polimeni, J. and Schwartz, E. L. (2001). *Space-time adaptive image representations: Data Structures, Hardware and Algorithms*.

Pollard, S. B., Porrill, J., Mayhew, J. E., and Frisby, J. P. (1986). Disparity gradient, lipschitz continuity, and computing binocular correspondences. In *Robotics Research: The Third International Symposium*, volume 30, pages 19–26. Massachusetts: MIT Press.

Posch, C., Matolin, D., and Wohlgenannt, R. (2010a). High-DR frame-free PWM imaging with asynchronous AER intensity encoding and focal-plane temporal redundancy suppression. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2430 –2433.

Posch, C., Matolin, D., and Wohlgenannt, R. (2010b). A QVGA 143db dynamic range asynchronous address-event PWM dynamic image sensor with lossless pixel-level video compression. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 400–401.

Posch, C., Matolin, D., and Wohlgenannt, R. (2011). A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS. *IEEE Journal of Solid-State Circuits*, 46(1):259–275.

Posch, C., Serrano-Gotarredona, T., Linares-Barranco, B., and Delbruck, T. (2014). Retinomorphic Event-Based Vision Sensors: Bioinspired Cameras With Spiking Output. *Proceedings of the IEEE*, 102(10):1470–1484.

Prazdny, K. (1985). Detection of binocular disparities. *Biological Cybernetics*, 52(2):93–99.

Preston, T. J., Li, S., Kourtzi, Z., and Welchman, A. E. (2008). Multivoxel Pattern Selectivity for Perceptually Relevant Binocular Disparities in the Human Brain. *The Journal of Neuroscience*, 28(44):11315–11327.

Prince, S. J. D., Cumming, B. G., and Parker, A. J. (2002). Range and mechanism of encoding of horizontal disparity in macaque V1. *Journal of Neurophysiology*, 87(1):209–221.

Prince, S. J. D. and Eagle, R. A. (2000). Weighted directional energy model of human stereo correspondence. *Vision Research*, 40(9):1143–1155.

Prince, S. J. D., Pointon, A. D., Cumming, B. G., and Parker, A. J. (2000). The Precision of Single Neuron Responses in Cortical Area V1 during Stereoscopic Depth Judgments. *The Journal of Neuroscience*, 20(9):3387–3400.

Qian, N. (1994). Computing Stereo Disparity and Motion with Known Binocular Cell Properties. *Neural Computation*, 6(3):390–404.

Qian, N. and Andersen, R. A. (1997). A physiological model for motion-stereo integration and a unified explanation of Pulfrich-like phenomena. *Vision Research*, 37(12):1683–1698.

Qian, N. and Zhu, Y. (1997). Physiological computation of binocular disparity. *Vision Research*, 37(13):1811–1827.

Qiao, N., Mostafa, H., Corradi, F., Osswald, M., Stefanini, F., Sumislawska, D., and Indiveri, G. (2015). A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128k synapses. *Frontiers in Neuroscience*, 9.

Qiu, F. T. and von der Heydt, R. (2005). Figure and Ground in the Visual Cortex: V2 Combines Stereoscopic Cues with Gestalt Rules. *Neuron*, 47(1):155–166.

Quam, L. H. (1984). Hierarchical Warp Stereo. Technical report.

Ramón y Cajal, S. .-. (1909). *Histologie du système nerveux de l'homme et des vertébrés*.

Rasche, C. and Hahnloser, R. H. R. (2001). Silicon synaptic depression. *Biological Cybernetics*, 84(1):57–62.

Read, J. C. A. and Cumming, B. G. (2005). The stroboscopic Pulfrich effect is not evidence for the joint encoding of motion and depth. *Journal of Vision*, 5(5):3–3.

Read, J. C. A. and Cumming, B. G. (2007). Sensors for impossible stimuli may solve the stereo correspondence problem. *Nature Neuroscience*, 10(10):1322–1328.

Regan, D. (1993). Binocular correlates of the direction of motion in depth. *Vision Research*, 33(16):2359–2360.

Regan, M. J. P., Miller, G. S. P., Rubin, S. M., and Kogelnik, C. (1999). A Real-time Low-latency Hardware Light-field Renderer. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 287–290, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

Rhemann, C., Hosni, A., Bleyer, M., Rother, C., and Gelautz, M. (2011). Fast cost-volume filtering for visual correspondence and beyond. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3017–3024.

Rieke, F. and Rudd, M. E. (2009). The challenges natural images pose for visual adaptation. *Neuron*, 64(5):605–616.

Rogister, P., Benosman, R., Ieng, S.-H., Lichtsteiner, P., and Delbruck, T. (2012). Asynchronous Event-Based Binocular Stereo Matching. *Neural Networks and Learning Systems, IEEE Transactions on*, 23(2):347 –353.

Rohaly, A. M. and Wilson, H. R. (1994). Disparity averaging across spatial scales. *Vision Research*, 34(10):1315–1325.

Rokers, B., Cormack, L. K., and Huk, A. C. (2009). Disparity- and velocity-based signals for three-dimensional motion perception in human MT+. *Nature Neuroscience*, 12(8):1050–1055.

Rovere, G., Ning, Q., Bartolozzi, C., and Indiveri, G. (2014). Ultra low leakage synaptic scaling circuits for implementing homeostatic plasticity in neuromorphic architectures. In *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2073–2076.

Roy, S. and Cox, I. J. (1998). A Maximum-Flow Formulation of the N-Camera Stereo Correspondence Problem. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 492–, Washington, DC, USA. IEEE Computer Society.

Rusinkiewicz, S. and Levoy, M. (2001). Efficient variants of the ICP algorithm. In *Third International Conference on 3-D Digital Imaging and Modeling, 2001. Proceedings*, pages 145–152.

Sarpeshkar, R. (1998). Analog versus digital: extrapolating from electronics to neurobiology. *Neural computation*, 10(7):1601–1638.

Sasaki, K. S., Tabuchi, Y., and Ohzawa, I. (2010). Complex Cells in the Cat Striate Cortex Have Multiple Disparity Detectors in the Three-Dimensional Binocular Receptive Fields. *The Journal of Neuroscience*, 30(41):13826–13837.

Scharstein, D. (1994). Matching images by comparing their gradient fields. In *, Proceedings of the 12th IAPR International Conference on Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing*, volume 1, pages 572–575 vol.1.

Scharstein, D. (1999). *View Synthesis Using Stereo Vision*. Springer-Verlag, Berlin, Heidelberg.

Scharstein, D. and Pal, C. (2007). Learning Conditional Random Fields for Stereo. In *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, pages 1–8.

Scharstein, D. and Szeliski, R. (1998). Stereo Matching with Nonlinear Diffusion. *International Journal of Computer Vision*, 28(2):155–174.

Scharstein, D. and Szeliski, R. (2002). A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *International Journal of Computer Vision*, 47(1-3):7–42.

Scharstein, D. and Szeliski, R. (2003). High-accuracy stereo depth maps using structured light. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings*, volume 1, pages I–195–I–202 vol.1.

Schemmel, J., Fieres, J., and Meier, K. (2008). Wafer-scale integration of analog neural networks. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 431–438. IEEE.

Schemmel, J., Grübl, A., Meier, K., and Mueller, E. (2006). Implementing synaptic plasticity in a VLSI spiking neural network model. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, pages 1–6. IEEE.

Schor, C., Wood, I., and Ogawa, J. (1984). Binocular sensory fusion is limited by spatial resolution. *Vision Research*, 24(7):661–665.

Schraml, S., Belbachir, A., and Brandle, N. (2010a). A real-time pedestrian classification method for event-based dynamic stereo vision. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 93–99.

Schraml, S., Belbachir, A., Milosevic, N., and Schön, P. (2010b). Dynamic stereo vision system for real-time tracking. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1409–1412.

Seitz, P. (1989). Using Local Orientational Information As Image Primitive For Robust Object Recognition. volume 1199, pages 1630–1639.

Serrano-Gotarredona, R., Oster, M., Lichtsteiner, P., Linares-Barranco, A., Paz-Vicente, R., Gomez-Rodriguez, F., Camunas-Mesa, L., Berner, R., Rivas-Perez, M., Delbruck, T., Liu, S.-C., Douglas, R., Hafliger, P., Jimenez-Moreno, G., Ballcels, A., Serrano-Gotarredona, T., Acosta-Jimenez, A., and Linares-Barranco, B. (2009). CAVIAR: A 45k Neuron, 5m Synapse, 12g Connects/s AER Hardware Sensory #x2013;Processing #x2013; Learning #x2013;Actuating System for High-Speed Visual Object Recognition and Tracking. *IEEE Transactions on Neural Networks*, 20(9):1417–1438.

Serrano-Gotarredona, R., Serrano-Gotarredona, T., Acosta-Jiménez, A., and Linares-Barranco, B. (2006). A neuromorphic cortical-layer microchip for spike-based event processing vision systems. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 53(12):2548–2566.

Serrano-Gotarredona, T. and Linares-Barranco, B. (2013). A 128 128 1.5% Contrast Sensitivity 0.9% FPN 3 us Latency 4 mW Asynchronous Frame-Free Dynamic Vision Sensor Using Transimpedance Preamplifiers. *Solid-State Circuits, IEEE Journal of*, 48(3):827–838.

Serrano-Gotarredona, T., Park, J., Linares-Barranco, A., Jimenez, A., Benosman, R., and Linares-Barranco, B. (2013). Improved contrast sensitivity DVS and its application to event-driven stereo vision. In *2013 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2420–2423.

Shapley, R. and Enroth-Cugell, C. (1984). Visual adaptation and retinal gain controls. *Progress in retinal research*, 3:263–346.

Sheik, S., Chicca, E., and Indiveri, G. (2012). Exploiting device mismatch in neuromorphic VLSI systems to implement axonal delays. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 1–6. IEEE.

Shimonomura, K., Kushima, T., and Yagi, T. (2008). Binocular robot vision emulating disparity computation in the primary visual cortex. *Neural Networks*, 21(2–3):331–340.

Shimonomura, K. and Yagi, T. (2005). A multichip aVLSI system emulating orientation selectivity of primary visual cortical cells. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 16(4):972–979.

Shioiri, S., Saisho, H., and Yaguchi, H. (2000). Motion in depth based on inter-ocular velocity differences. *Vision Research*, 40(19):2565–2572.

Simoni, M. F., Cymbalyuk, G. S., Sorensen, M. E., Calabrese, R. L., and DeWeerth, S. P. (2004). A multiconductance silicon neuron with biologically matched dynamics. *Biomedical Engineering, IEEE Transactions on*, 51(2):342–354.

Sivilotti, M. A. (1990). *Wiring considerations in analog VLSI systems, with application to field-programmable networks*. PhD thesis, Citeseer.

Smallman, H. S. (1995). Fine-to-coarse scale disambiguation in stereopsis. *Vision Research*, 35(8):1047–1060.

Stephan Schraml, P. S. (2007). Smartcam for real-time stereo vision - address-event based embedded system. pages 466–471.

Stevenson, S. B., Cormack, L. K., Schor, C. M., and Tyler, C. W. (1992). Disparity tuning in mechanisms of human stereopsis. *Vision Research*, 32(9):1685–1694.

Sulzbachner, C., Zinner, C., and Kogler, J. (2011). An optimized Silicon Retina stereo matching algorithm using time-space correlation. In *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–7.

Sun, J., Li, Y., Kang, S. B., and Shum, H.-Y. (2005). Symmetric stereo matching for occlusion handling. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*, volume 2, pages 399–406 vol. 2.

Sun, J., Zheng, N.-N., and Shum, H.-Y. (2003). Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):787 – 800.

Szeliski, R. (1990). Bayesian modeling of uncertainty in low-level vision. *International Journal of Computer Vision*, 5(3):271–301.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. (2008). A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080.

Takemura, A., Inoue, Y., Kawano, K., Quaia, C., and Miles, F. A. (2001). Single-Unit Activity in Cortical Area MST Associated With Disparity-Vergence Eye Movements: Evidence for Population Coding. *Journal of Neurophysiology*, 85(5):2245–2266.

Tanabe, S., Doi, T., Umeda, K., and Fujita, I. (2005). Disparity-Tuning Characteristics of Neuronal Responses to Dynamic Random-Dot Stereograms in Macaque Visual Area V4. *Journal of Neurophysiology*, 94(4):2683–2699.

Tanabe, S., Haefner, R. M., and Cumming, B. G. (2011). Suppressive Mechanisms in Monkey V1 Help to Solve the Stereo Correspondence Problem. *The Journal of Neuroscience*, 31(22):8295–8305.

Tanabe, S., Umeda, K., and Fujita, I. (2004). Rejection of False Matches for Binocular Correspondence in Macaque Visual Cortical Area V4. *The Journal of Neuroscience*, 24(37):8170–8180.

Tanaka, H., Uka, T., Yoshiyama, K., Kato, M., and Fujita, I. (2001). Processing of Shape Defined by Disparity in Monkey Inferior Temporal Cortex. *Journal of Neurophysiology*, 85(2):735–744.

Terzopoulos, D. (1986). Regularization of Inverse Visual Problems Involving Discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(4):413–424.

Thomas, O. M., Cumming, B. G., and Parker, A. J. (2002). A specialization for relative disparity in V2. *Nature Neuroscience*, 5(5):472–478.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-E., Koelen, C., Markey, C., Rummel, C., Niekerk, J. v., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., and Mahoney, P. (2007). Stanley: The Robot That Won the DARPA Grand Challenge. In Buehler, M., Iagnemma, K., and Singh, S., editors, *The 2005 DARPA Grand Challenge*, number 36 in Springer Tracts in Advanced Robotics, pages 1–43. Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-73429-1_1.

Tippetts, B., Lee, D. J., Lillywhite, K., and Archibald, J. (2013). Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, pages 1–21.

Toumazou, C., Lidgey, F. J., and Haigh, D. (1990). *Analogue IC design: the current-mode approach*, volume 2. Presbyterian Publishing Corp.

Trivedi, H. P. and Lloyd, S. A. (1985). The role of disparity gradient in stereo vision. *Perception*, 14(6):685–690.

Tsai, J. J. and Victor, J. D. (2003). Reading a population code: a multi-scale neural model for representing binocular disparity. *Vision Research*, 43(4):445–466.

Tsang, E. K. C. and Shi, B. E. (2004). A preference for phase-based disparity in a neuromorphic implementation of the binocular energy model. *Neural Computation*, 16(8):1579–1600.

Tschechne, S., Sailer, R., and Neumann, H. (2014). Bio-inspired optic flow from event-based neuromorphic sensor input. In *Artificial Neural Networks in Pattern Recognition*, pages 171–182. Springer.

Uka, T. and DeAngelis, G. C. (2004). Contribution of Area MT to Stereoscopic Depth Perception: Choice-Related Response Modulations Reflect Task Strategy. *Neuron*, 42(2):297–310.

Uka, T. and DeAngelis, G. C. (2006). Linking Neural Representation to Function in Stereoscopic Depth Perception: Roles of the Middle Temporal Area in Coarse versus Fine Disparity Discrimination. *The Journal of Neuroscience*, 26(25):6791–6802.

Van Schaik, A. and Jin, C. (2003). The tau-cell: a new method for the implementation of arbitrary differential equations. In *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on*, volume 1, pages I–569. IEEE.

Varela, J. A., Sen, K., Gibson, J., Fost, J., Abbott, L. F., and Nelson, S. B. (1997). A quantitative description of short-term plasticity at excitatory synapses in layer 2/3 of rat primary visual cortex. *The Journal of neuroscience*, 17(20):7926–7940.

Veksler, O. (1999). *Efficient Graph-based Energy Minimization Methods in Computer Vision*. PhD thesis, Cornell University, Ithaca, NY, USA. AAI9939932.

Veksler, O. (2001). Stereo matching by compact windows via minimum ratio cycle. In *Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings*, volume 1, pages 540–547 vol.1.

Vogelstein, R. J., Mallik, U., Vogelstein, J. T., and Cauwenberghs, G. (2007). Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses. *Neural Networks, IEEE Transactions on*, 18(1):253–265.

Wang, L., Yang, R., Gong, M., and Liao, M. (2012). Real-time stereo using approximated joint bilateral filtering and dynamic programming. *Journal of Real-Time Image Processing*, 9(3):447–461.

Wang, Z.-F. and Zheng, Z.-G. (2008). A region based stereo matching algorithm using cooperative optimization. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, pages 1–8.

Weikersdorfer, D., Hoffmann, R., and Conradt, J. (2013). Simultaneous Localization and Mapping for Event-Based Vision Systems. In Chen, M., Leibe, B., and Neumann, B., editors, *Computer Vision Systems*, number 7963 in Lecture Notes in Computer Science, pages 133–142. Springer Berlin Heidelberg.

Wijekoon, J. H. and Dudek, P. (2008). Compact silicon neuron circuit with spiking and bursting behaviour. *Neural Networks*, 21(2):524–534.

Witkin, A., Terzopoulos, D., and Kass, M. (1987). Signal matching through scale space. *International Journal of Computer Vision*, 1(2):133–144.

Xue, J. T., Ramoa, A. S., Carney, T., and Freeman, R. D. (1987). Binocular interaction in the dorsal lateral geniculate nucleus of the cat. *Experimental Brain Research*, 68(2):305–310.

Yang, Q. (2014). Hardware-Efficient Bilateral Filtering for Stereo Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):1026–1032.

Yang, Q., Wang, L., Yang, R., Stewenius, H., and Nister, D. (2009). Stereo Matching with Color-Weighted Correlation, Hierarchical Belief Propagation, and Occlusion Handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):492–504.

Yang, Q., Wang, L., Yang, R., Wang, S., Liao, M., and Nister, D. (2006). Real-time Global Stereo Matching Using Hierarchical Belief Propagation. In *British Machine Vision Conference*.

Yao, E., Hussain, S., Basu, A., and Huang, G.-B. (2013). Computation using mismatch: Neuromorphic extreme learning machines. In *Biomedical Circuits and Systems Conference (BioCAS), 2013 IEEE*, pages 294–297. IEEE.

Yoon, K.-J. and Kweon, I. S. (2006). Adaptive support-weight approach for correspondence search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):650–656.

Yu, T., Park, J., Joshi, S., Maier, C., and Cauwenberghs, G. (2012). 65k-neuron integrate-and-fire array transceiver with address-event reconfigurable synaptic routing. In *2012 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 21–24.

Yu, W., Chen, T., Franchetti, F., and Hoe, J. (2010). High Performance Stereo Vision Designed for Massively Data Parallel Platforms. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1509–1519.

Yuille, A. L. and Poggio, T. (1984). A Generalized Ordering Constraint for Stereo Correspondence.

Zabih, R. and Woodfill, J. (1994). Non-parametric local transforms for computing visual correspondence. In Eklundh, J.-O., editor, *Computer Vision — ECCV '94*, number 801 in Lecture Notes in Computer Science, pages 151–158. Springer Berlin Heidelberg.

Zaghloul, K., Boahen, K., and others (2004a). Optic nerve signals in a neuromorphic chip I: Outer and inner retina models. *Biomedical Engineering, IEEE Transactions on*, 51(4):657–666.

Zaghloul, K., Boahen, K., and others (2004b). Optic nerve signals in a neuromorphic chip II: Testing and results. *Biomedical Engineering, IEEE Transactions on*, 51(4):667–675.

Zhang, K., Lu, J., Yang, Q., Lafruit, G., Lauwereins, R., and Van Gool, L. (2011). Real-Time and Accurate Stereo: A Scalable Approach With Bitwise Fast Voting on CUDA. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(7):867–878.

Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999*, volume 1, pages 666–673 vol.1.

Zicari, P., Perri, S., Corsonello, P., and Cocorullo, G. (2012). Low-cost FPGA stereo vision system for real time disparity maps calculation. *Microprocessors and Microsystems*, 36(4):281–288.

Zitnick, C. and Kanade, T. (2000). A cooperative algorithm for stereo matching and occlusion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):675–684.

Zitnick, C. L. and Kang, S. B. (2007). Stereo for Image-Based Rendering using Image Over-Segmentation. *International Journal of Computer Vision*, 75(1):49–65.

Curriculum Vitae

Marc Osswald

Personal Information

Name:	Marc Osswald
Address:	Schaffhauserstrasse 199
	8057 Zürich
Nationality:	Swiss
Mobile phone:	(+41) 079 228 32 89
Email:	marc@ini.uzh.ch
Date of birth:	27th February 1987
Place of birth:	Basel, Switzerland

Education

06/2011-12/2015 **Institute of Neuroinformatics,** ETHZ and University of Zurich, Switzerland
PhD in Neuromorphic Engineering under the supervision of Prof. G. Indiveri
Neuromorphic Processors: Event-based VLSI models of cortical circuits for brain-inspired computation

09/2008-03/2011 **Swiss Federal Institute of Technology (ETHZ),** Zurich, Switzerland
MSc in Mechanical Engineering with specialization in Robotics and Control Systems

10/2005-08/2008 **Swiss Federal Institute of Technology (ETHZ),** Zurich, Switzerland
BSc in Electrical Engineering and Information Technology with specialization in Electrical Power Systems and Mechatronics

08/2000-07/2005 **Gymnasium Kirschgarten,** Basel, Switzerland
Qualification for university entrance (Matura), equivalent to High School with a Major in Physics and Mathematics and a Minor in Chemistry

Academic Experience

Spring/Fall 2013 **ETH Zurich, Institute of Neuroinformatics,** Switzerland
Preparation and supervision of the Neuromorphic Engineering I&II classes as a **teaching assistant**.

09/2008-05/2009 **ETH Zurich, Automatic Control Laboratory,** Switzerland
Supervising laboratory training classes for students as a **teaching assistant**.

Research Experience

06/2011-12/2015 **PhD Thesis**, University of Zurich, Institute of Neuroinformatics
"Event-based Neuromorphic Stereo Vision"

09/2013-12/2013 **Research fellow**, Institut de la Vision, Paris, France
Collaboration with the "Natural Vision and Computation" group. Development of event-based algorithms for stereo vision.

09/2010-03/2011 **Master thesis**, ETH Zurich, Bio-Inspired Robotics Laboratory
"A Climbing Robot based on Hot Melt Adhesion (HMA)"

Spring 2009 **Semester thesis**, ETH Zurich, Automatic Control Laboratory
"Real-time tracking and control of 1:43 scale race cars"

Spring 2007 **Semester project**, ETH Zurich, Power Electronic Systems Laboratory
Design and implementation of a PI controlled DC/DC converter for a solar-powered car.

Work Experience

Since 05/2014 **Insightness**, Zurich, Switzerland
Co-Founder and Head of Product
Management duties and responsibilities. Particularly involved in product management. Development of visual positioning systems (VPS) prototypes.

09/2009-03/2010 **ESA, European Astronaut Center (EAC),** Cologne, Germany
Development of hardware and software for the upgrade of the Russian Soyuz Spacecraft Simulator. Astronaut Basic Training Support: Attending Basic Training lessons as student and observer, supporting development of lessons and providing technical support for Basic Training.

01/2009-09/2009 **ETH Zurich, Automatic Control Laboratory**, Switzerland
Setting up experiments as a **research assistant** for the student's mandatory laboratory training class: Implementation of an industrial controller for a two-degree of freedom helicopter in collaboration with the company B&R Automation (Austria).

Part-time Job Experience

09/2012-12/2015 **Academic Sports Association Zurich (ASVZ),** Switzerland
Soccer coach and head coach of the University's soccer team.

Publications

- Qiao, N., Mostafa, H., Corradi, F., Osswald, M., Stefanini, F., Sumislawska, D., & Indiveri, G. (2015). A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses. Frontiers in neuroscience, 9.
- Mostafa, H., Corradi, F., Osswald, M., & Indiveri, G. (2013, September). Automated synthesis of asynchronous event-based interfaces for neuromorphic systems. In Circuit Theory and Design (ECCTD), 2013 European Conference on (pp. 1-4). IEEE.

- Osswald, M., & Iida, F. (2013). Design and control of a climbing robot based on hot melt adhesion. Robotics and Autonomous Systems, 61(6), 616-625.
- Osswald, M., & Iida, F. (2011, September). A climbing robot based on hot melt adhesion. In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on (pp. 5107-5112). IEEE.

Patents

- Brändli, C., Osswald, M. & Delbrück, T. "Method for tracking keypoints in a scene", EP14178447.0 (pending)

Languages

German	native language
English	very good in speech and writing – equivalent to level C1
French	good in speech and writing – equivalent to level B1

Computer Skills

Operating systems	Ubuntu, Windows, Mac OS X
Programming	C/C++, Java, Matlab, VHDL
Software	Cadence, Xilinx ISE, LabVIEW, Altium Designer, NX, SolidWorks, Microsoft Office, Latex, Inkscape, Photoshop

Activities and certifications

03/2013-07/2013	Fitness Instructor Certificate, Swiss Academy of Fitness & Sports (SAFS), Zurich, Switzerland
01/2010-04/2010	PADI Open Water Diver, ESA and DLR Scuba Diving Center, Cologne, Germany

Hobbies

Various sports including soccer, beach volleyball, mountain biking, cycling, ski mountaineering and hiking. Furthermore I am a passionate photographer and movie maker.